Cognitive Assistants for Evidence-Based Reasoning Tasks

Mihai Boicu, Dorin Marcu, Gheorghe Tecuci, David Schum

Learning Agents Center, George Mason University, Fairfax, VA 22030

Abstract

Evidence-based reasoning is at the core of many problem solving and decision making tasks in a wide variety of domains. This paper introduces a computational theory of evidence-based reasoning, the architecture of a learning agent shell which incorporates general knowledge for evidence-based reasoning, a methodology that uses the shell to rapidly develop cognitive assistants in a specific domain, and a sample cognitive assistant for intelligence analysis.

Introduction

Evidence-based reasoning is at the core of many problem solving and decision making tasks in a wide variety of domains, including law, intelligence analysis, forensics, medicine, physics, chemistry, history, archaeology, and many others.

"In its simplest sense, evidence may be defined as any factual datum which in some manner assists in drawing conclusions, either favorable or unfavorable, to some hypothesis whose proof or refutation is being attempted" (Murphy, 2003, p.1). The conclusions are probabilistic in nature because our evidence is always *incomplete* (we can look for more, if we have time), usually *inconclusive* (it is consistent with the truth of more than one hypothesis), frequently *ambiguous* (we cannot always determine exactly what the evidence is telling us), commonly *dissonant* (some of it favors one hypothesis but other evidence favors other hypotheses), and has various degrees of *believability* (Schum, 2001a; Boicu et al., 2008). Often stunningly complex *arguments*, requiring both *imaginative and critical reasoning* through *deduction*, *induction*, and *abduction*, are necessary in order to establish and defend the three major credentials of evidence: its *relevance*, its *believability*, and its *inferential force or weight* with respect to the considered hypotheses (Tecuci et al., 2010b).

Because evidentiary reasoning is frequently of such an astonishing complexity, we believe that it can be best approached through the mixed-initiative integration of human imagination and computer knowledge-based reasoning (Tecuci et al., 2007). Toward this purpose, the next section introduces several elements of an emerging computational theory of evidence-based reasoning, as a basis for automating parts of this process. This includes general knowledge, such as an ontology of evidence and believability assessment procedures. This theory, as well as previous work on Disciple learning assistants (Tecuci, 1998; Tecuci et al., 2001, 2002, 2005; Boicu 2002), is at the basis of a learning agent shell that includes a general knowledge base for evidence-based reasoning. Using this agent shell, a subject matter expert can rapidly develop a cognitive assistant for a specific domain, with limited knowledge engineering support, by teaching it domain-specific knowledge and reasoning, which extends the agent's general knowledge for evidence-based reasoning.

After introducing the general architecture of the shell, the paper presents a methodology for rapid development of a specific cognitive assistant, illustrating it with an example of a cognitive assistant for intelligence analysis.

Toward a Computational Theory of Evidence-based Reasoning

Evidence-based Reasoning as Discovery of Evidence, Hypotheses, and Arguments

We view evidence-based reasoning as collaborative processes of evidence in search of hypotheses, hypotheses in search of evidence, and evidential tests of hypotheses, all taking place simultaneously, as illustrated in the upper right part of Figure 1. Through *abductive reasoning*, which shows that something is *possibly* true (Peirce, 1898), we generate hypotheses from our observations; through *deductive reasoning*, which shows that something is *necessarily* true, we use our hypotheses to generate new lines of inquiry and discover new evidence; and through *inductive reasoning*, which shows that something is *probably* true, we test our hypotheses with this discovered evidence. The left and bottom part of Figure 1 illustrate this approach with an example from intelligence analysis.

Imagine that a counterterrorism analyst reads in today's

Washington Post an article where a person named Willard reports that a canister of cesium-137 has gone missing from the XYZ Company in Maryland (see E at the bottom-left of Figure 1). The question is: What hypothesis of interest would explain this evidence? To answer it, the analyst and her cognitive assistant develop the following chain of abductive inferences (see the left side of Figure 1).

E: There is evidence that the cesium-137 canister is missing → H_1: It is possible that the cesium-137 canister is indeed missing → H_2: It is possible that it was stolen → H_3: It is possible that it was stolen by someone associated with a terrorist organization → H_4: It is possible that the terrorist organization will use the cesium-137 canister to build a dirty bomb → H_5: It is possible that the dirty bomb will be set off in the Washington DC area.

In this case, we have *evidence in search of hypotheses* where an item of evidence "searches" for hypotheses that explain it.

But these are not the only hypotheses that explain the evidence. Indeed, just because there is evidence that the cesium-137 canister is missing does not mean that it is indeed missing. At issue here is the believability of Willard who provided this information. Thus an alternative hypothesis is that the cesium-137 canister is not missing. But let us assume that it is missing. Then it is possible that it was stolen. But it is also possible that it was lost, or maybe it was used in a project at the XYZ Company.

The analyst and her cognitive assistant need to assess each of these hypotheses, starting from bottom-up, by employing the general approach from the upper right of Figure 1. So they put hypothesis H_1 at work to guide them in collecting additional evidence, as illustrated by the middle (blue) tree at the bottom of Figure 1. The question is: Assuming that the hypothesis H_1 is true, what other things should be observable? The reasoning might start as follows: "If H_1 were true, there are sub-hypotheses, listed as H_{11}, H_{12}, and H_{13}, that would be necessary and sufficient (or, at least, sufficient) to make H_1 true. If H_{11} were true then one would need to observe evidence that supports it. Similarly, H_{12} and H_{13} lead to the deduction of potential items of evidence (shown as shaded circles) that bear upon them. In our example, the hypothesis "H_1: The cesium-137 canister is missing from the XYZ Warehouse" is reduced to the following three sub-hypotheses: "H_{11}: The cesium-137 canister is registered as being in the XYZ warehouse"; "H_{12}: The cesium-137 canister is not found in the XYZ warehouse"; and "H_{13}: The cesium-137 canister was not checked out from the XYZ warehouse." As a result, the analyst contacts Ralph, the supervisor of the warehouse, who reports that the cesium-137 canister is registered as being in the warehouse, that no one at the XYZ Company had checked it out, but it is not located anywhere in the hazardous materials locker. He also indicates that the lock on the hazardous materials locker appear to have been forced. Ralph's testimony provides several items of

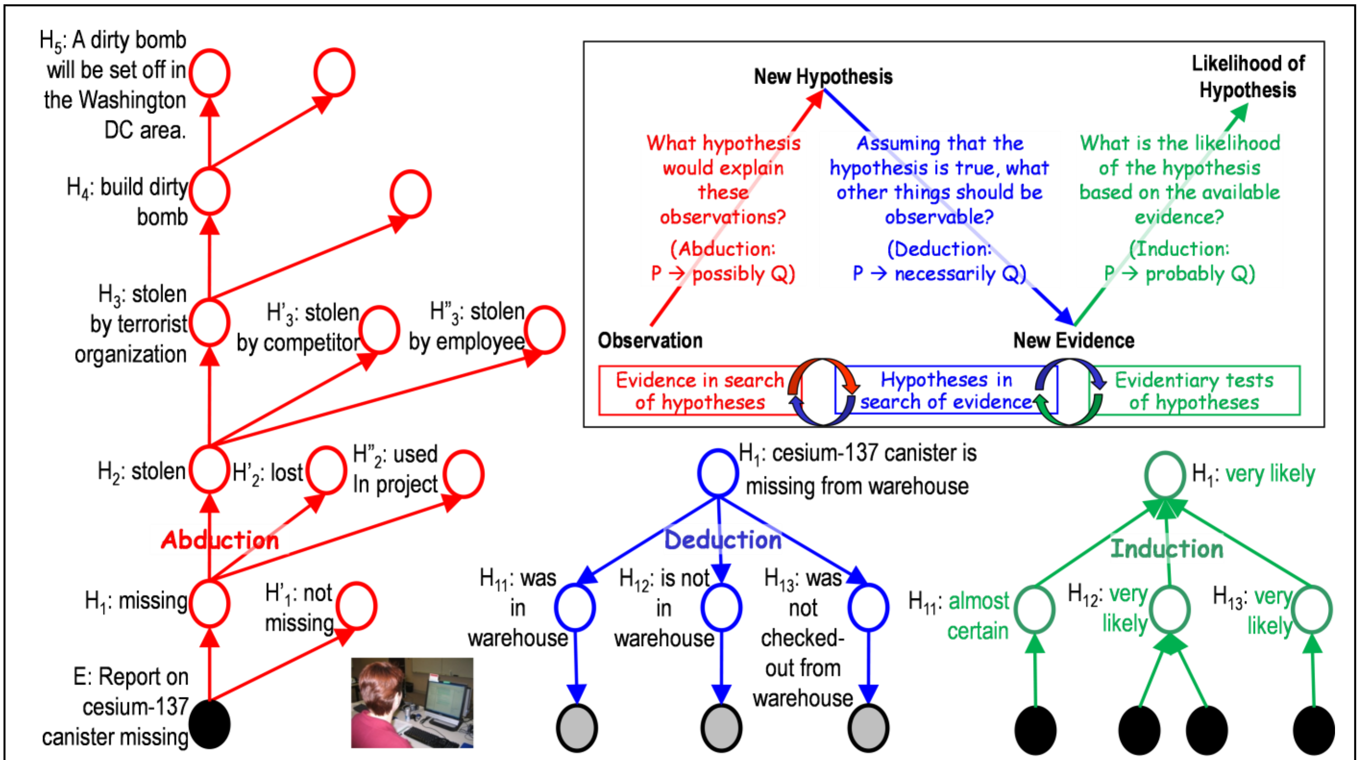

Figure 1: Evidence-based hypothesis generation and analysis.

relevant evidence in support of H_{11}, H_{12}, and H_{13}.

So here we have hypothesis in search of evidence that guides the analyst in collecting new evidence.

Now, some of the newly discovered items of evidence may trigger new hypotheses, or the refinement of the current hypotheses (e.g. "H_2: The cesium-137 canister was stolen from the hazardous material locker"). So, as indicated by the curved arrows at the bottom of the abstract diagram in Figure 1, the processes of evidence in search of hypotheses and hypotheses in search of evidence take place at the same time, and in response to one another.

The combination of evidence in search of hypotheses and hypotheses in search of evidence results in hypotheses which have to be tested, through *inductive reasoning*, based on the discovered items of evidence, as illustrated at the bottom-right part of Figure 1. This is a probabilistic inference network that shows how the evidence at the leaves of the network (shown as black circles) is linked to the hypotheses H_{11}, H_{12}, H_{13}, and to H_1, through an argument that establishes the relevance, believability and inferential force of evidence with respect to these hypotheses, as discussed in the next section. The result of the testing process is the likelihood of the hypothesis H_1: It is very likely that the cesium-137 canister is missing from the XYZ warehouse.

Having concluded that the cesium-137 canister is missing, the analyst and her cognitive assistant now need to consider the competing hypotheses H_2, H'_2, and H''_2, from the left part of Figure 1. Each of these hypotheses will be put to work to guide the analyst in collecting new items of evidence, and the identified evidence is used to assess the likelihoods of these competing hypotheses. If, for instance, it is determined that the cesium-137 canister was used in a project at the XYZ Company, then the investigation stops here. But if it is determined that the cesium-137 canister was stolen, then the analyst and the cognitive assistant will continue with the investigation of upper level hypotheses (i.e., H_3, H'_3, H''_3, …).

While this is a systematic approach to hypotheses generation and analysis, except for trivial cases, it is too complex to be done manually. Indeed, real cases of analysis involve hundreds or thousands of items of evidence. We therefore claim that it can and has to be performed by analysts supported by their cognitive assistants, as discussed in the rest of this paper.

Knowledge-based Hybrid Reasoning

Notice that the investigative strategy that we have just illustrated is a type of hybrid reasoning where small abduction, deduction, and induction steps feed on each other, in a recursive spiral of discovery of hypotheses, evidence, and arguments that provide a better and better understanding of the world.

Another important aspect is that the analysis of an upper level hypothesis (e.g. "H_2: stolen") makes use of the previously performed analyses of the hypotheses on the chain from the initial evidence E to it (i.e., "H_1: missing"). That is, both the middle (blue) and the right (green) trees corresponding to "H_2: stolen" will contain, as subtrees, the middle and the right trees corresponding to "H_1: missing," shown at the bottom of Figure 1.

Our goal is to enable a cognitive assistant to perform much of this reasoning, under the guidance of a human user. Figure 2 shows how the generated hypotheses are assessed by employing a divide and conquer approach (called *problem reduction and solution synthesis*) which combines the deductive and inductive reasoning trees from Figure 1. This approach is grounded in the problem reduction representations developed in the field of artificial intelligence (Nilsson, 1971; Tecuci, 1998), and in the argument construction methods provided by the noted jurist John H. Wigmore (1937), the philosopher of science Stephen Toulmin (1963), and the evidence professor David Schum (1987, 2001a). In essence, a hypothesis to be assessed is successively reduced to simpler and simpler hypotheses. Then the simplest hypotheses are assessed based on the available evidence, resulting in probabilistic solutions because of the characteristics of evidence mentioned in the introduction. Finally, these probabilistic solutions are successively combined, from bottom-up, to obtain the solution of the top-level hypothesis.

In Figure 2 the assessment of the hypothesis H_1 is reduced to the assessment of three simpler hypotheses, H_{11}, $H_{12,}$ and H_{13}. The middle hypothesis H_{12} is assessed based on the available evidence. As indicated in Figure 2, one has to consider both *favoring evidence* and *disfavoring evidence*. In this illustration there are two items of favoring evidence, E_1 and E_2. Therefore one has to assess to what extent each of these items of evidence favors the hypothesis H_{12}. This, in turn, requires the assessment of the relevance of E_1 to H_{12}, and the believability of E_1 which, in this illustration, are assumed to be: "If we believe E_1 then H_{12} is almost certain" and "It is likely that E_1 is true." By composing the assessed relevance and believability of E_1 (e.g., through a "min" function) one assesses the inferential force of E_1 on H_{12} ("Based on E_1 it is likely that H_{12} is true"). Similarly one assesses the inferential force of E_2 on H_{12} ("almost certain"). Then, by composing these solutions (e.g., through a "max" function), one assesses the inferential force or weight of the favoring evidence (i.e. both E_1 and E_2) on H_{12}: Based on the favoring evidence it is almost certain that H_{12} is true. Through a similar process one assesses the inferential force of the disfavoring evidence on H_{12}, and then the likelihood of H_{12}, based both on the favoring and the disfavoring evidence ("It is very likely that H_{12} is true"). H_{11} and H_{13} are assessed in a similar way. Then the assessments of H_{11}, H_{12}, and H_{13} are combined into the assessment of the top level hypothesis H_1: "It is very likely that H_1 is true."

Notice that the reduction and synthesis tree from Figure 2 combines the deduction (blue) tree and the induction (green) tree from the bottom right of Figure 1. Notice also that we have expressed the probabilistic solutions by using symbolic probability values (e.g. "very likely") which, in the current implementation correspond to specific probability intervals, as shown in the table from the left of Figure 2.

A basic element of the computational theory of evidence-based reasoning is a "substance-blind" ontology of evidence (Boicu et al., 2008; Schum, 2009) which is applicable in every domain. A fragment of this ontology is shown in the bottom right of Figure 2. This ontology distinguishes between various types of *tangible evidence* and *testimonial evidence*, where each type is associated with a specific believability evaluation procedure, as illustrated at the left of this ontology fragment. For example, to assess the believability of an item of *unequivocal testimonial evidence based upon direct observation* (such as Ralph's testimony mentioned in the previous section), one needs to assess the *competence* and the *credibility* of the source (i.e. Ralph). Competence involves *access* (Did Ralph actually make the observation he claims to have made? Did he have access to the information he reports?) and *understandability* (Did Ralph understand what he observed well enough to provide us with an intelligible account?). Credibility involves *veracity* (Is Ralph telling us about an event he believes to have occurred?), *objectivity* (Did Ralph base his belief on sensory evidence received during an observation, or did he believe the reported event occurred either because he expected or wished it to occur?), and *observational sensitivity under the conditions of observation* (If Ralph did base a belief on sensory evidence, how good was this sensory evidence?). This knowledge is based upon 600 years of experience in our system of law (Schum, 2001a),

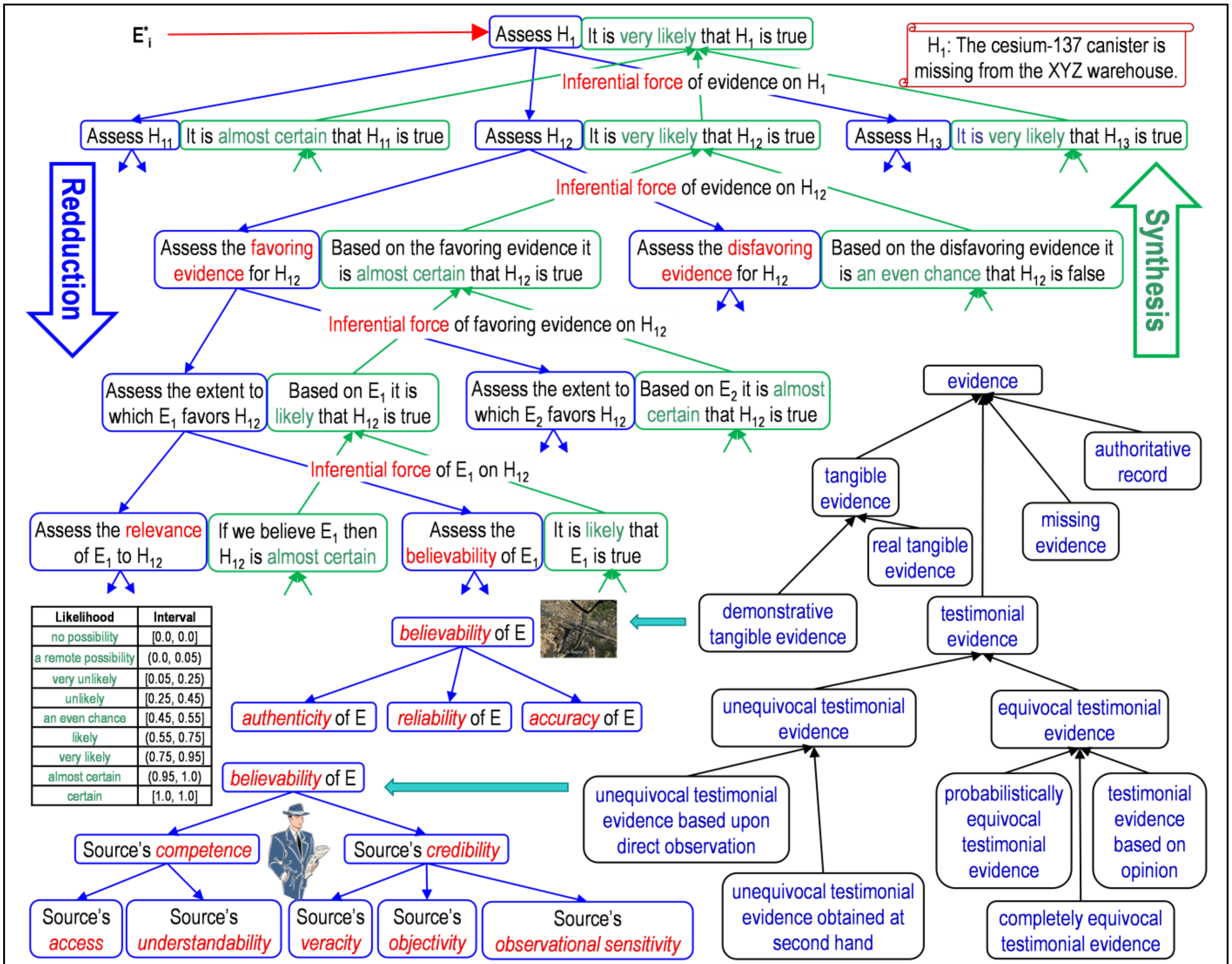

Figure 2: Hypothesis analysis through problem reduction and solution synthesis.

and is incorporated in the knowledge base of the agent.

Notice that the reasoning approach illustrated in Figure 2 represents a *natural integration of logic and probabilistic reasoning.* This is important because most of the work on the development of cognitive systems has been done either in a logical framework or a probabilistic one, with not much interaction between the two research communities.

One problem in evidence-based reasoning is that there are several quite different views among probabilists about what the force or weight of evidence means and how it should be assessed and combined across evidence in both simple and complex arguments, such as Subjective Bayes, Baconian, Belief Functions, and Fuzzy (Schum, 2001a). This is primarily because none of these views can cope with all the five characteristics of evidence mentioned in the introduction, when testing a hypothesis. For example, Subjective Bayes, which is by far the best known and most frequently used, cannot cope with the incompleteness of evidence. In fact, only the Baconian view, which is the least known, can cope with this. But neither the Subjective Bayes view nor the Baconian view can cope with the ambiguity in evidence. However, both the Belief Functions view and the Fuzzy view have solutions for this.

While here we have primarily illustrated the use of fuzzy-like probabilities, Baconian elements appear in several places. One is that each sub-hypothesis of a hypothesis, such as the sub-hypotheses H_{11}, H_{12}, and H_{13} of H_1, is, in fact, a test for H_1. Moreover, if we do not have evidence for a sub-hypothesis (for example H_{13}), we can give it the benefit of the doubt and reason "as if" the hypothesis H_{13} were true. Our goal, of course, is to synergistically integrate more of the existing probability views within a logical argumentation structure, to provide more informative assessments of our hypotheses.

Sample Evidence-based Reasoning Tasks

The examples provided in the previous sections are based on the TIACRITIS cognitive assistant for intelligence analysts which is currently experimentally used in several government organizations (Tecuci et al., 2010a, b). We claim, however, that this type of evidence-based reasoning is applicable for many tasks. We have developed, for example, another cognitive assistant for modeling the behavior of violent extremists, and a cognitive assistant for assessing a potential PhD advisor.

Scientists from various domains, such as physics, chemistry, or biology, may regard the presented approach as a formulation of the scientific method (Noon, 2009), and we have developed a case study that uses it to teach the scientific method to middle school students.

In law, a prosecutor makes observations in a criminal case and seeks to generate hypotheses in the form of charges that seem possible in explaining these observations. Then, assuming that a charge is justified, attempts are made to deduce further evidence bearing on it. Finally, the obtained evidence is used to prove the charge.

In medicine, a doctor makes observations with respect to a patient's complaints and attempts to generate possible diagnoses (hypotheses) that would explain them. She then performs medical tests that provide further evidence which is used in forming a final diagnosis for the patient.

In forensics (Noon, 2009), observations made at the site of an explosion in a power plant lead to the formulation of several possible causes. The analysis of each possible cause leads to the discovery of new evidence that eliminates or refines some of the causes, and may even suggest new ones. This cycle continues until enough evidence is found to determine the most likely cause.

Learning Agent Shell for EBR Tasks

The cognitive assistant for evidence-based reasoning briefly presented above was developed by employing the Disciple learning agent shell for evidence-based reasoning. As indicated in Figure 3, this shell includes multiple modules for problem solving, learning, tutoring, evidence-based reasoning, mixed-initiative interaction, as well as a hierarchically organized knowledge base with domain-independent knowledge for evidence-based reasoning at the top of the knowledge hierarchy. A Disciple shell can learn complex problem solving expertise directly from human experts and, in doing so, it evolves into a cognitive assistant that can support experts and non-experts in problem solving, and can teach their problem solving expertise to students.

Over the years, the knowledge representation, problem solving, and learning methods of the Disciple cognitive assistants have been continuously improved, as they have been applied to many domains, including course of action critiquing (Tecuci et al., 2001), military center of gravity determination (Tecuci et al., 2002), and, most recently, intelligence analysis (Tecuci et al., 2010a,b). A newer feature of this shell is that now it also includes general knowledge for evidence-based reasoning from the emerging Science of Evidence (Schum, 2009). Therefore, what remains to be acquired from an expert is specific domain knowledge. Moreover, this knowledge is rapidly acquired by employing its learning methods. Maintenance and adaptation is done through additional learning.

In essence, the knowledge base of the agent consists of an *evolving ontology of concepts* and different types of *if-then rules expressed with these concepts*. The ontology includes both general concepts for evidence-based reasoning, such as the partial evidence ontology from the bottom-right of Figure 2, and domain-specific concepts. It is assumed that the ontology is incomplete and is

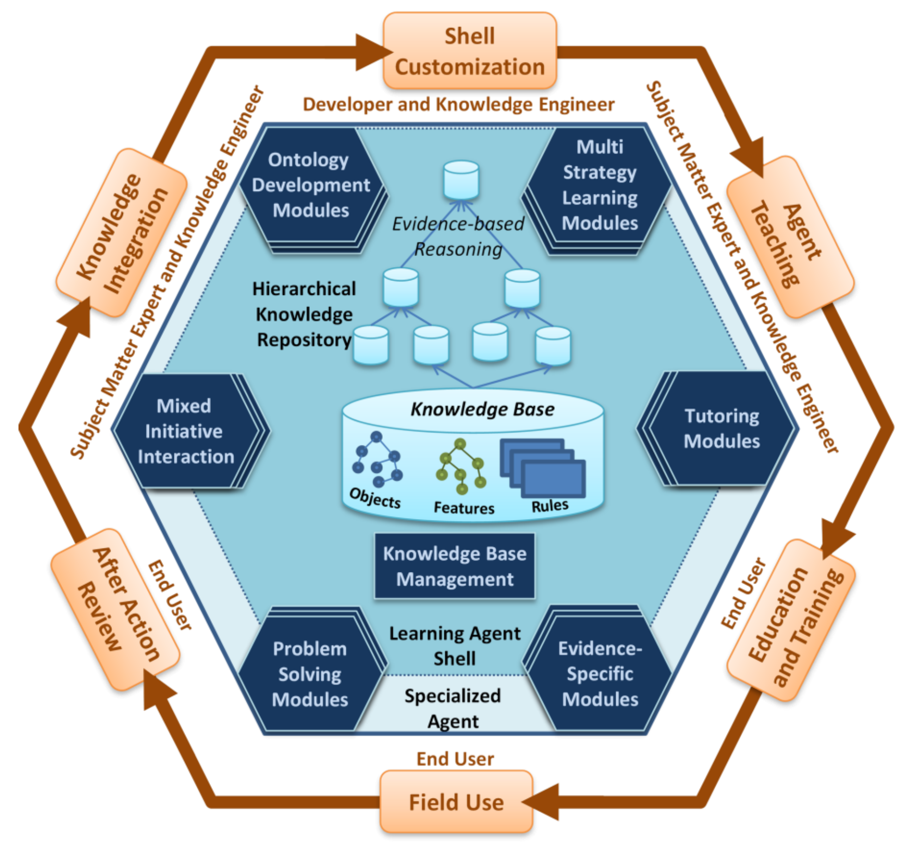

Figure 3: The general architecture and life cycle of an evidence-based cognitive assistant.

continuously extended with new concepts and features, through knowledge acquisition and learning (Boicu, 2002).

An *if-then problem reduction rule* expresses how and under what conditions a generic hypothesis can be reduced to simpler generic hypotheses. These conditions are represented as first-order logical expressions (Tecuci et al., 2005). It is by the application of such rules that an agent can generate the reduction part of the tree in Figure 2. Similarly, an *if-then solution synthesis rule* expresses how and under what conditions generic probabilistic solutions can be combined into another probabilistic solution. The application of these rules generates the synthesis part of the tree in Figure 2.

A Disciple agent is taught directly by a subject matter expert, with limited knowledge engineering assistance, in a way that is similar to how one would teach a student or apprentice, through specific examples and explanations, and through the supervision and correction of its problem solving behavior (Tecuci et al., 2005). In essence, a subject matter expert will show the agent a sample reasoning tree like the ones in Figure 1, and the agent will learn general rules and ontology elements from the specific reasoning steps and their explanations, with the help of the expert. This integrates teaching and multistrategy learning, where the expert helps the agent to learn (e.g., by providing examples, hints and explanations), and the agent helps the expert to teach it (e.g., by asking relevant questions and proposing plausible explanations).

The left part of Figure 4 shows a reduction step provided by the subject matter expert, where a specific problem P_1 is reduced to n subproblems P_{11}, … , P_{1n}. To help the expert express her reasoning in this way, we instruct her to formulate a question Q on how to solve P_1, and then provide the answer A that should guide her to reduce P_1 to n subproblems. In this way, the Q/A pair represents the reason for this reduction, or its explanation.

To learn a general reduction rule from this specific reduction, the agent needs first to understand the meaning of the Q/A pair in terms of the object concepts and relationships from its ontology. This meaning is illustrated by the network fragment at the bottom left of Figure 4 that shows the objects O_1, O_2, and O_3 and the relationships f_1 and f_2 between them. Based on the reduction and its explanation, an initial rule is automatically generated by replacing each object from the reduction with a general variable, and by adding a main applicability condition restricting the possible values of these variables. The general structure of the learned rule is shown in the right part of Figure 4. Initially, the rule only contains the main applicability condition (shown in green), but no except-when conditions (shown in red), or exceptions. The main applicability condition is represented as a plausible version space consisting of a plausible lower bound condition and

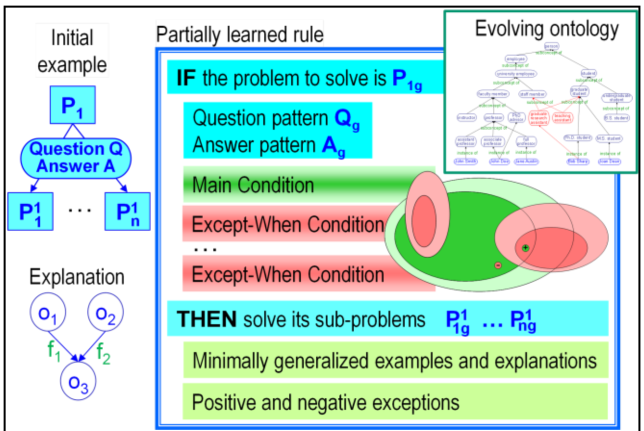

Figure 4: Learning with an evolving representation language.

a plausible upper bound condition. They are obtained as minimal and, respectively, maximal generalizations of the objects from the reduction example, in the context of the current ontology which is used a generalization hierarchy.

As the agent learns new rules and concepts from the expert, their interaction evolves from a teacher-student interaction, toward an interaction where they both collaborate in problem solving. In this case the agent automatically generates parts of the reasoning tree and the expert critiques the reasoning, implicitly guiding the agent in refining its rules. For example, based on the explanation of why an instance of the rule in Figure 4 is wrong, the agent learns an except-when plausible version space condition which is added to the rule. Such conditions should not be satisfied in order to apply the rule. Correct reductions may lead to the generalization of the plausible lower bound of the main applicability condition, or to the specialization of the plausible upper bound of one or several except-when condition, or to the addition of a positive exception when none of the above operations is possible. Incorrect reductions and their explanations may lead to the specialization of the plausible upper bound of the main applicability condition, or to the generalization of the plausible lower bound of an except-when condition, or to the learning of a plausible version space for a new except-when condition, or to the addition of 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 with new object concepts and features. For example, to explain the agent why a generated reduction is wrong, the expert may use a new object concept or feature. As a result, the agent will add the new concept or a new feature definition 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 the rules in the context of a changed ontology. Notice that this is, in fact, a form of learning with an evolving representation language.

The way a partially learned rule is used depends on the current goal of the agent. If this goal is to support its user in problem solving, then the agent will generate the reductions that are more likely to be correct, such as a reduction covered by the plausible lower bound of the main condition and not covered by any of the Except-When plausible upper bound conditions. However, if the current goal of the agent is to improve its reasoning rules, then it will generate reductions that will speed up the rule refinement process, such as a reduction that is covered by the plausible upper bound of the main condition, or the plausible upper bound of an except-when condition.

A challenging issue is supporting the user in generating new hypotheses through abductive or imaginative reasoning. One important feature of the presented approach is that a hypothesis of interest (e.g., “A dirty bomb will be set off in the Washington DC area”) is not generated through a single glorious abduction from evidence. Instead, it emerges from the spiral hybrid reasoning that generates chains of simpler abductions from evidence to hypotheses which are increasingly more mission-relevant. To generate these simpler abductions, we research an approach that combines ideas from Thagard’s (1993) simple, existential, and analogical abductions, with ideas from Eco’s (1983) overcoded, undercoded, and creative abductions. In our approach each step is a combined abduction, such as *simple undercoded abduction*, or *analogical overcoded abduction*, as first proposed in (Schum, 2001b).

Agent Lifecycle

The outside border hexagon in Figure 3 summarizes the life cycle of a Disciple cognitive assistant for evidence-based reasoning. The first stage is *Shell Customization* where, based on the specification of the type of problems to be solved and the agent to be built, the developer and the knowledge engineer may decide that some extensions of the Disciple shell may be necessary or useful. It is through such successive extensions during the development of Disciple agents for various applications that the current version of the shell for evidence-based reasoning problems (which includes the EBR knowledge base) has emerged.

The next stage is *Agent Teaching* by subject matter expert and the knowledge engineer, supported by the agent itself which simplifies and speeds-up the knowledge base development process (Tecuci et al., 2001; 2002; 2005).

Once an operational agent is developed, it is used for the *Education and Training* of the end users, possibly in a classroom environment. The next stage is *Field Use*, were copies of these agents support users in their operational environments. In this stage an agent assists its user both in solving problems and in collaborating with other users and their cognitive assistants. At the same time, it continuously learns from this problem solving experience by employing a form of non-disruptive learning. In essence, it learns new rules from examples, as illustrated in Figure 4. However, because there is no assistance from the user, the learned rules will not include a formal applicability condition. It is during the next stage of *After Action Review*, when the user and the agent analyze past problem solving episodes, that the formal applicability conditions are learned, based on the accumulated examples.

In time, each cognitive assistant extends its knowledge with expertise acquired from its user. This creates the opportunity of developing a more competent agent by integrating the knowledge of all these agents. This is accomplished by a knowledge engineer, with assistance from a subject matter expert, in the next stage of *Knowledge Integration*. The result is an improved agent which may be used in a new iteration of a continuous process of spiral development and use.

The TIACRITIS agent, developed with the Disciple shell, is a web agent specialized for teaching intelligence analysts the critical thinking skills needed to perform evidence-based reasoning (Tecuci et al., 2010a, b). The agent is accompanied by a textbook that teaches basic knowledge about the properties, uses, and marshaling of evidence to show students what is involved in assessing the relevance, believability, and inferential force credentials of evidence. It includes a wide array of examples of the use of the TIACRITIS agent and hands-on exercises involving both real and hypothetical cases chosen to help students recognize and evaluate many of the complex elements of the analyses they are learning to perform. Each chapter starts with a presentation of some important matter, such as assessing the believability of evidence. Then the students are asked to use TIACRITIS and experiment with what they have just been taught. Both the textbook and the agent are easily customizable by selecting the chapters and the case studies to be used.

A just-developed agent for modeling the behavior of violent extremists includes the entire knowledge base for evidence-based reasoning of TIACRITIS.

Conclusions

We have presented a learning agent shell that enables rapid development of specific cognitive assistants for tasks requiring evidence-based reasoning. In addition to having the necessary knowledge representation, problem solving, and learning modules, it also includes general knowledge for evidence-based reasoning. Thus, to evolve into a cognitive assistant for a specific domain, it only needs domain-specific knowledge, which is acquired through learning from a subject matter expert.

Acknowledgements

This research was performed in the Learning Agents Center and was supported by George Mason University and by several agencies of the U.S. Government, including the National-Geospatial Intelligence Agency, and the Department of Defense. The U.S. Government is authorized to reproduce and distribute reprints notwithstanding any copyright notation thereon.

References

Boicu, M. (2002). *Modeling and Learning with Incomplete Knowledge*, PhD Thesis in Information Technology, Learning Agents Center, George Mason University.

Boicu, M., Tecuci, G., Schum, D.A. (2008). Intelligence Analysis Ontology for Cognitive Assistants, in *Proc. of the Conference "Ontology for the Intelligence Community,"* VA, 3-4 December.

Eco, U. (1983). Horns, Hooves, Insteps: Some Hypotheses on Three Types of Abduction. In Eco, U. and Sebeok, T. (eds.), *The Sign of Three: Dupin, Holmes, Peirce*, pp.198-220, Indiana University Press: Bloomington.

Murphy, P. (2003). *Evidence, Proof, and Facts: A Book of Sources.* Oxford University Press, Oxford, UK.

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

Noon, R.K. (2009). *Scientific Method: Applications in Fact Investigation and Forensic Science.* CRC Press, Taylor & Francis, Boca Raton, Florida.

Peirce, C.S. (1898). *Reasoning and the Logic of Things*, 1898. In Ketner K., ed., Harvard University Press, Cambridge, MA, 1992.

Schum, D.A. (1987). *Evidence and Inference for the Intelligence Analyst* (2 Volumes). Lanham, MD: University Press of America.

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

Schum, D.A. (2001b). Species of Abductive Reasoning in Fact Investigation in Law. *Cardozo Law Review*, 22(5-6), pp.1645–1681.

Schum, D.A. (2009). Science of Evidence: Contributions from Law and Probability. *Law Probab Risk* **8** 197–231.

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

Tecuci, G., Boicu, M., Bowman, M., Marcu, D., with commentary by Burke, M. (2001). An Innovative Application from the DARPA Knowledge Bases Programs: Rapid Development of a Course of Action Critiquer, *AI Magazine,* **22(2)**, 43-61.

Tecuci, G., Boicu, M., Marcu, D., Stanescu, B., Boicu, C., Comello, J. (2002). Training and Using Disciple Agents: A Case Study in the Military Center of Gravity Analysis Domain, in *AI Magazine,* **24(4)**, 51-68, AAAI Press, CA.

Tecuci, G., Boicu, M., Boicu, C., Marcu, D., Stanescu, B., Barbulescu, M. (2005) The Disciple-RKF learning and reasoning agent. *Computational Intelligence* **21(4)** 462–479.

Tecuci, G., Boicu, M., Cox, M.T. (2007). Seven Aspects of Mixed-initiative Reasoning: An Introduction to the Special Issue on Mixed-initiative Assistants. *AI Magazine* **28(2),** 11–18.

Tecuci, G., Schum D.A., Boicu, M., Marcu, D. (2010a), *Introduction to Intelligence Analysis: A Hands-on Approach with TIACRITIS,* 220 pg, George Mason University.

Tecuci, G., Schum, D.A., Boicu, M., Marcu, D., Hamilton, B., Wible, B. (2010b). Teaching Intelligence Analysis with TIACRITIS, *American Intelligence Journal,* **28**(2), pp. 50-65.

Thagard, P.R. (1993). *Computational Philosophy of Science*, MIT Press: Cambridge, MA.

Toulmin, S.E. (1963). *The Uses of Argument*. Cambridge U Press.

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