Published in FLAIRS'03 (2003 International Florida Artificial Intelligence Research Society Conference)

Case-Based Argumentation via Process Models

J. William Murdock, David W. Aha, & Leonard A. Breslow

Intelligent Decision Aids Group
Navy Center for Applied Research in Artificial Intelligence
Naval Research Laboratory, Code 5515
Washington, DC 20375
surname@aic.nrl.navy.mil

Abstract

We introduce AHEAD (Analogical Hypothesis Elaborator for Activity Detection), a software system we are developing for the DARPA EELD (Evidence Extraction and Link Discovery) program. AHEAD performs case-based hypothesis elaboration using process models. We are applying AHEAD, which embodies a domain-independent approach, to elaborate hypothesized hostile activities. In this application, AHEAD is given as input (1) structured evidence and (2) a hypothesis concerning the activities of an asymmetric adversary (e.g., a terrorist organization). The system outputs a detailed symbolic argument supporting and/or opposing the given hypothesis. Combining case-based/analogical reasoning techniques, using the FIRE Analogy Server (Forbus, 2001) with qualitative functional processes represented as Task-Method-Knowledge models (Stroulia & Goel, 1995; Goel & Murdock, 1996), AHEAD extracts additional implications of the hypothesis to develop a coherent argument that supports and/or contradicts it. We detail AHEAD's design, its role in EELD, its implementation status, and our plans for its evaluation.

Introduction

Asymmetric threats occur when a small, secretive (e.g., terrorist or criminal) group threatens a large, powerful (e.g., military or law enforcement) group. A key challenge in combating asymmetric threats is *detection*: determining what the asymmetric adversary is attempting. For example, if a law enforcement group detects an attempt by an organized crime group to take over a commercial industry in some region, the law enforcement group can then attempt to stop the takeover or reverse it. In many asymmetric threat domains data sets are both large and complex, involving many different types of relationships among entities. Consequently, detection can be exceedingly difficult, requiring an enormous amount of time to perform evidence analysis, pattern matching, and link discovery.

The DARPA Evidence Extraction and Link Discovery (EELD) program is creating software for automatic discovery of potential asymmetric threats. EELD consists of research and development in three primary areas: evidence extraction, link discovery, and pattern learning. *Evidence extraction* tools convert unstructured data (i.e., raw text) into structured data (e.g., semantic networks or databases). *Link discovery* tools match collections of structured data to known threat patterns. Finally, *pattern learning* discovers new patterns of asymmetric threats. The EELD program integrates these three areas, to perform fast and accurate asymmetric threat detection.

An integrated EELD system can generate many hypotheses of varying credibility. Consequently, an additional challenge arises, namely *elaboration*: providing information to help a user determine whether a hypothesized threat is genuine and decide how to respond to it. In this paper, we introduce AHEAD (Analogical Hypothesis Elaborator for Activity Detection), the EELD component that supports hypothesis elaboration. AHEAD does not extract data from text or detect threats from extracted data; it relies on other EELD systems to provide data and hypothesized threats. AHEAD receives as input a hypothesis from EELD's link discovery components, along with evidence used to create that hypothesis, and produces as output an argument for and/or against that hypothesis. These arguments should help a user (e.g., an intelligence analyst) to quickly and confidently decide how to respond to hypothesized asymmetric threats.

Motivation

One conceivable way to perform both detection and elaboration would be to use a *pure* case-based reasoning (CBR) approach. New evidence would be used to retrieve concrete instances of past threats. If the degree of match was high, the system would report to the user that it had detected a new situation similar to a previous threat situation, and details of this match could persuade the user that the new threat is genuine. This approach may be feasible in domains in which there are a large number of previously encountered threat instances and they include a substantial proportion of all possible threats (e.g., shoplifting, because countless examples occur every day, and the degree of variation among individual examples is fairly small). However, in many asymmetric threat domains (e.g., terrorism, organized crime), threats are relatively infrequent and are sufficiently complex that a virtually limitless range of variations exists. Thus any new threat that arises is unlikely to be an exact or near-exact match to some past instance and is therefore unlikely to be detected or elaborated through pure CBR (because CBR relies on adapting near matches) In our research we are focusing on domains of this sort.

Consequently, we are pursuing a multi-modal reasoning approach using multiple abstraction levels. Rather than using a case base composed of individual concrete instances, we are using process models as generalized cases (i.e., a single model encodes an abstract representation of a hostile process such as a takeover of an industry by an organized crime group; many individual instances of takeovers could match to a single model). Furthermore, we are employing model-based reasoning in addition to case-based reasoning. Systems that integrate CBR with other reasoning approaches are well represented in the AI literature (Marling *et al*., 2002), as are systems that employ generalizations of concrete cases (Bergmann & Wilke, 1996) and systems that use representations of process as cases (Tautz & Fenstermacher, 2001).

Sibyl (Eilbert, 2002) is an EELD project that uses CBR for hypothesis generation. Sibyl's cases are also generalizations of specific instances but closely resemble the evidence in structure and content. AHEAD's cases differ significantly from Sibyl's because they are used for different purposes that impose different demands. Whereas Sibyl focuses on searching for threats and uses cases that enable fast search through large bodies of evidence, AHEAD does not perform this search. Instead, it is given a hypothesis, which is directly linked to relevant pieces of evidence, and focuses on its elaboration. Thus, AHEAD's cases do not need to be structured for fast retrieval but do need to include information not only on *what kinds* of evidence are consistent with a given hypothesized threat, but also on *why* that evidence is consistent with it. Consequently, AHEAD uses *functional* models of processes as cases; such models describe not only the actions performed in the process but also how those actions contribute to the overall effect.

Although some previous CBR research projects have employed functional process models for explanation, they have not focused on arguments about detected processes. Instead, explanation via functional process models has been limited to explanation of a system's own reasoning (e.g., Goel & Murdock, 1996). AHEAD combines functional process models and case-based reasoning in a challenging new context.

While previous work has studied argumentation in CBR, that work focused on the use of concrete cases in domains where generalized cases are unavailable (e.g., Branting, 1990). AHEAD generates arguments using retrieved models generalized across a range of specific instances. Furthermore, AHEAD's arguments display an innovative structure, derived from the capabilities provided by functional process models and from the goal of helping a user understand a complex detected hypothesis.

Case Representation: TMK Models

AHEAD uses a case representation that includes information about how the process is performed and why portions of the process contribute to its overall objective. This representation is known as the TMK (Task-Method-Knowledge) modeling framework. A TMK model is divided into *tasks* (defining what the process is intended to accomplish), *methods* (defining how the process works), and *knowledge* (information that drives the process by providing context).

In AHEAD, TMK's focus on *purpose* enables reasoning about whether the actions of asymmetric adversaries support some hypothesized task that they are suspected of attempting. TMK has been extensively shown to provide useful information about temporal and causal relationships among aspects of a process. However, no prior work on TMK has investigated *organizational relationships* (e.g., how tasks and methods are divided among a variety of actors and how knowledge is exchanged among those actors). Also, no prior work on TMK has focused on building models of processes performed by *adversaries*; the information needed to support inferences about purposes and methods can differ significantly when the individual or group performing a process is deliberately obscuring their purposes and methods. In AHEAD, we are enhancing existing TMK formalisms to address these new issues.

Figure 1 displays a high-level overview of a sample TMK model in the AHEAD project. The rectangles represent tasks, the rounded boxes represent methods, and the oblique parallelograms represent parameters. Labeled links denote relational information encoded in the tasks and methods. These links connect tasks, methods, parameters, and other links. For example, there is a link labeled *makes* from the *Industry-Takeover* task to the link labeled *controls* from the *Mafia* parameter to the *Target-Industry* parameter. Those links indicate that an industry takeover produces a state in which the mafia controls the target industry. At the bottom of Figure 1 are ellipses, indicating that those tasks can be further decomposed by additional methods into lower-level tasks.

There are many reasons why the knowledge used to

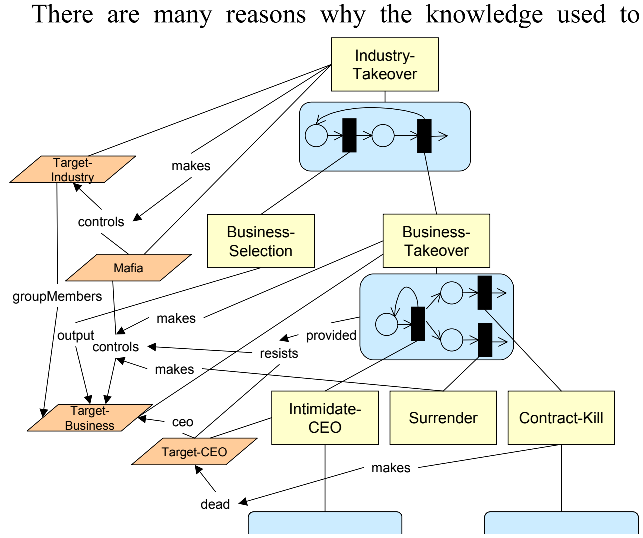

Figure 1: Overview of an example TMK model

create a hypothesis is not appropriate for producing an argument for why the hypothesis is valid. Some hypothesis generation techniques use almost completely incomprehensible knowledge (e.g., artificial neural networks), while others use knowledge that may be comprehensible in small pieces but lack the organization and abstraction needed to produce a complete, comprehensible argument (e.g., decision trees and Bayesian networks). Even a hypothesis generator that directly matches evidence to a symbolic pattern provides only limited explanatory capabilities. It may be possible to show how some pieces of evidence match the pattern; this provides a partial argument. However, a pattern constructed (either automatically or by hand) for use in matching should contain extensive information about what elements correspond to various conclusions but will generally not contain any insights as to why these correspondences occur. Because TMK is a functional process modeling language (i.e., it encodes not only the elements of the process but also the *purposes* that those elements serve in the context of the process as a whole), an argument based on a TMK model not only indicates which pieces of evidence are consistent with the given hypothesis, but also identifies *why* that evidence supports the hypothesis. Consequently, TMK is well suited to addressing AHEAD's knowledge requirements.

Acquisition of TMK models is a key issue for AHEAD. The existing TMK models have been constructed by hand based on domain information supplied by the developers of the EELD challenge problem. As we scale up to a wide variety of real-world asymmetric threat domains, we will need increasingly powerful tools and procedures for obtaining models. Fortunately, there is a large body of existing work dealing with human-driven knowledge acquisition for process information (e.g., Abowd *et al.*, 1997; Althoff *et al.*, 2002) on which to build as AHEAD progresses.

Approach

Given a hypothesis and its supporting evidence, AHEAD creates an argument for and/or against the hypothesis. Figure 2 displays an overview of AHEAD's approach. The behavior of AHEAD is divided into three phases: process identification, trace extraction, and argument generation. These phases are based on the inputs and the case library (i.e., the collection of TMK models). The output is an argument, presented to the user via a graphical user interface. The individual phases are described below:

Process identification: The first phase relates the given hypothesis to a TMK model. One aspect of this phase is *case retrieval*: identifying which model in AHEAD's library of TMK models is most relevant to the given hypothesis. The other aspect of the process identification phase is *analogical mapping*: relating specific portions of the hypothesis to corresponding portions of the retrieved model. In principle, these two aspects could be implemented as separate steps. However, AHEAD instead

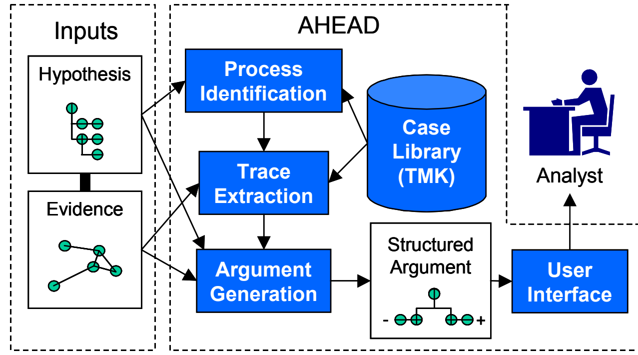

Figure 2: Functional architecture for AHEAD

uses an off-the-shelf analogical mapping system that performs integrated retrieval and mapping

The analogical mapping system that AHEAD uses is the FIRE Analogy Server (from the Institute for Learning Sciences' Qualitative Reasoning Group at Northwestern University), a general-purpose analogical reasoning tool (Forbus, 2001). The portion of the Analogy Server that AHEAD directly invokes is MAC/FAC (Gentner & Forbus, 1991), which selects the case (i.e., a TMK model) that most closely matches the input hypothesis and identifies a mapping between elements of the input and the case.

Consider, for example, the following hypothesis: a local organized crime group has taken over a given industry in a given city and has killed two people during that takeover. AHEAD would invoke MAC/FAC to select a model matching that hypothesis from the case library. MAC/FAC would retrieve a TMK model of industry takeovers and map its parameters to entities in the hypothesis (e.g., the parameter for target industry would be mapped to the industry mentioned in the hypothesis). If no model exactly matches the type of activity being performed, MAC/FAC can retrieve an approximate match. For example, if AHEAD receives a hypothesis regarding organized crime takeover of postal service in some area, MAC/FAC could recognize that the overall structure of the hypothesis resembles the structure of industry takeovers, even though a postal service is a government organization, not an industry. The specific entities in the hypothesis can then be mapped to analogous model elements (e.g., the local postal service is not a target industry but it is analogous to one). AHEAD would then perform trace extraction and argument generation using this partially relevant model.

Trace extraction: In the second phase, AHEAD constructs a *trace* (i.e., a permissible temporal path) through the matched TMK model that is consistent with the evidence. Because the trace constitutes an elaboration of the hypothesized case, trace extraction is the *solution proposal* portion of the general CBR process (Kolodner & Leake, 1996). To illustrate, if an input hypothesis posits an industry takeover, the trace extraction process should use the mapping between the specific hypothesis and a general model of industry takeovers to determine a temporal path through the model that could produce the observed evidence. Insights from this process include inferences about what parts of the model have been completed and what parts are underway (e.g., that one company in the industry is being threatened but has not yet been taken over). Each part of the trace includes direct links to any pieces of evidence that either support or contradict the conclusion that the specified part of the trace did occur. AHEAD can quickly find the relevant evidence because the input hypothesis is already linked to the input evidence on which the hypothesis is based.

For example, the model of industry takeovers (Figure 1) involves attempts to take over multiple businesses within the industry. A business takeover is further decomposed into lower level tasks of intimidating a CEO and killing the CEO. There are multiple paths through the model. For example, if the criminal organization succeeds in taking over a business after intimidating a CEO, then it has no need to kill that CEO, so AHEAD would not include a step in the trace that encodes a killing. However, if the crime group failed to intimidate the CEO but did not kill the CEO, then the trace would contain a step that represents the killing (because the model states that it should occur) along with evidence that the step did not occur.

Argument Generation: Finally, AHEAD constructs arguments concerning the hypothesis using the extracted trace. Generation of an argument constitutes the *justification* step of the general process for CBR (Kolodner & Leake, 1996). More specifically, the argument generation process steps through the extracted trace and produces arguments for and/or against the input hypothesis based on the evidence

For example, there can be evidence from a business takeover suggesting that a group intimidated the CEO, did not take over the business, and did not kill the CEO. In this example, one portion of the argument AHEAD produces would support the hypothesis of an industry takeover (because intimidating a CEO is part of industry takeovers), while another portion of the argument would contradict the hypothesis (because killing the CEO would be expected under the circumstances but did not occur). A user examining the argument could decide that the latter evidence is strong enough to conclude that an industry takeover has not occurred. Alternatively, the user might conclude that the crime group simply acted in an atypical manner or that the activity is still taking place. AHEAD does not draw conclusions of these sorts; it simply presents the relevant arguments to the user.

AHEAD Output: Arguments

The output of AHEAD is the arguments it produces (displayed in the GUI). AHEAD's argumentation structure is inspired by Toulmin (1958), who describes an argument as a link between data and a claim in the form of a warrant (an abstract assertion that relates the data to the claim) supported by backing (additional information that implies the validity of a warrant) and qualified by a rebuttal (an assertion that limits the scope of the claim). The core elements of Toulmin's argument formalism (the data and the claim) map directly to the inputs of AHEAD (data=evidence, claim=hypothesis). However, significant differences exist between the intermediate elements of our formalism and Toulmin's. These differences are largely due to differences in the use of arguments in AHEAD versus the purposes of arguments that Toulmin considers.

For example, Toulmin focuses primarily on arguments in which data and claims are both atomic assertions. In contrast, the evidence and hypotheses provided to AHEAD are complex and have multiple abstraction levels (e.g., an industry takeover is decomposed into takeovers of individual businesses, which are themselves further decomposed). Toulmin's arguments support the link between the data and the claim as an abstract rule ("warrant") and assume that the audience understands how that abstract rule applies. However, in AHEAD, the abstractions that can support the claims (i.e., the TMK models) must be non-atomic because the data and claims they support are non-atomic. Consequently, AHEAD generates *instantiated warrants*. For example, a Toulmin-style warrant in the industry takeover example might be that, when a crime group intimidates a CEO who still resists the takeover, the criminals will generally kill the CEO. A corresponding instantiated warrant in AHEAD would state that a specific crime group killed the CEO of a specific business because the CEO resisted a takeover.

Furthermore, Toulmin describes arguments that are intended to persuade someone that a claim is valid. In AHEAD, we are instead trying to help a user understand the strengths and weaknesses of a claim. The only counter-arguments ("rebuttals") that Toulmin examines are ones that restrict the claim's generality, rather than ones that contradict the claim. AHEAD does not need rebuttals of the sort Toulmin describes because the hypotheses that it takes as input involve concrete assertions, not generalizations. However, AHEAD does need to present information that opposes the claim. We refer to this information as *instantiated counter-warrants*, because they serve the same role as warrants but in support of the opposing position. One can view a structured argument in AHEAD as a dialog between two opposing sides. The elements of this dialog are, themselves, smaller arguments, which (like Toulmin's arguments) are intended to support a specific position. Each instantiated warrant or counter-warrant plus its data constitutes one such smaller argument. The combination of these individual arguments into the full structured argument is intended to help a user evaluate the strengths and weaknesses of the hypothesis.

AHEAD presents both the input hypotheses and their generated arguments to the user via its graphical user interface (GUI). The interface provides a variety of views, and enables navigation and analysis among a set of hypotheses and related arguments. Both the hypotheses and the arguments are presented using natural-language text. Figure 3 presents a screen shot showing the GUI displaying a sample hypothesis and argument.

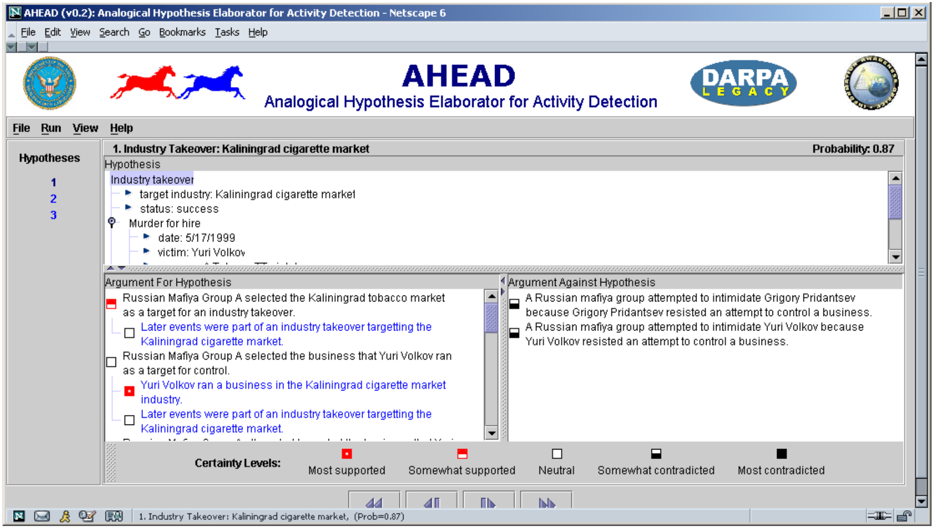

Figure 3: Screenshot of AHEAD's GUI

Evaluation

AHEAD is still under development; the trace extractor and parts of the argument generator are not yet operational. However, we have conducted a pilot user study focusing on AHEAD's GUI and arguments. This evaluation found preliminary evidence that arguments in AHEAD can help a user to understand hypotheses *faster*, to produce *more accurate* judgments of the hypotheses, and to be *more confident* in their judgments. The details of this evaluation will be presented in an upcoming paper (Murdock, Aha, & Breslow, 2003).

Acknowledgements

This research is sponsored by the Defense Advanced Research Projects Agency. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied of the Defense Advanced Research Projects Agency or the United States Government.

References

Abowd, G., Goel, A.K., Jerding, D., McCracken, M., Moore, M., Murdock, J.W., Potts, C., Rugaber, S., & Wills, L. (1997). *Proceedings of the International Conference on Software Maintenance* (pp. 150-159). Bari, Italy: IEEE Press.

Althoff, K.-D., Becker-Kornstaedt, U., Decker, B., Klotz, A., Leopold, E., Rech, J., & Voss, A. (2002). The indiGo project: Enhancement of experience management and process learning with moderated discourses. In P. Perner (Ed.), *data mining in marketing and medicine*. New York: Springer.

Bergmann, R., & Wilke, W. (1996). On the role of abstraction in case-based reasoning. *Proceedings of the Third European Workshop on Case-Based Reasoning* (pp. 28-43). Lausanne, Switzerland: Springer.

Branting, K. (1990). *Integrating rules and precedents for classification and explanation: Automating legal analysis*. Doctoral dissertation, University of Texas, Austin.

Eilbert, J. (2002, October). *Socio-culturally oriented plan discovery environment (SCOPE)*. Presentation at the Fall 2002 EELD PI Meeting, San Diego, CA.

Forbus, K. (2001). Exploring analogy in the large. In D. Gentner,, K. Holyoak, & B. Kokinov (Eds.) *The analogical mind: Perspectives from cognitive science* (pp. 23-58) Cambridge, MA: MIT Press.

Gentner, D., & Forbus, K. (1991). MAC/FAC: A model of similarity-based retrieval. *Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society* (pp. 504-509). Chicago, IL: Lawrence Erlbaum.

Goel, A.K., & Murdock, J.W. (1996). Meta-cases: Explaining case-based reasoning. *Proceedings of the Third European Workshop on Case-Based Reasoning* (pp. 150-163). Lausanne, Switzerland: Springer.

Kolodner, J., & Leake, D. (1996). A tutorial introduction to case-based reasoning. In D. Leake (Ed.) *Case-based reasoning: Experiences, lessons, & future directions* (pp. 31-65). Cambridge, MA: MIT Press & AAAI Press.

Marling, C., Sqalli, M., Rissland, E., Muñoz-Avila, H., & Aha, D.W. (2002). Case-based reasoning integrations. *AI Magazine*, *23*(1), 69-86.

Murdock, J. W., Aha, D.W., & Breslow, L.A. (2003). *Assessing elaborated hypotheses: An interpretive case-based reasoning approach*. Unpublished manuscript.

Stroulia, E., & Goel, A.K. (1995). Functional representation and reasoning in reflective systems. *Applied Intelligence*, 9, 101-124.

Tautz, C. & Fenstermacher, K. (Eds.) (2001). Case-base reasoning approaches for process-oriented knowledge management. In R. Weber & C.G. von Wangenheim (Eds.) *Proceedings of the Workshop Program at the Fourth International Conference on Case-Based Reasoning* (pp. 6-28). Vancouver, Canada: Naval Research Laboratory.

Toulmin, S. (1958). *The uses of argument*. Cambridge, UK: Cambridge University Press.