Instructing a Cognitive Agent to Perform Sensemaking in Intelligence, Surveillance and Reconnaissance

Gheorghe Tecuci TECUCI@GMU.EDU
Dorin Marcu DMARCU@GMU.EDU
Mihai Boicu MBOICU@GMU.EDU
Louis Kaiser LKAISER4@GMU.EDU
Learning Agents Center, George Mason University, Fairfax, VA 22030 USA

Abstract

This paper presents progress made towards the development of a theory, methodology and instructable system for automated analysis in Intelligence, Surveillance and Reconnaissance. An intelligence analyst shows the system how to make sense of a suspicious alert by following the scientific method of hypothesis generation and testing, and the system learns general rules to make sense of similar alerts. We believe that the approach we are developing has several significant advantages over the current manual approach to sensemaking in terms of speed, quality, and transparency of analysis, which more than compensate for the extra effort needed to train the agent. In particular, it significantly improves and accelerates the understanding of the goals and behavior of entities of interest, and enables the early identification of potential threats.

1. Introduction

Intelligence, Surveillance, and Reconnaissance (ISR) synchronizes and integrates the planning and the operation of collection assets, processing, exploitation, and dissemination systems in direct support of current and future operations (JP 1-02, 2010). Sensemaking is the process of situational understanding based on data that is sparse, noisy, and uncertain (Moore, 2011).

There is a huge gap between the ability to collect information and the ability to analyze and make sense of it. Sensemaking in ISR and, in general, in the field of intelligence analysis is performed mostly manually by analysts. The prevailing approach is a holistic analysis where the analysts, after reviewing large amounts of information and performing all the reasoning in their heads, reach a conclusion.

A complementary approach uses structured analytic techniques, such as those described by Heuer and Pherson (2011) that guide the hypothesis generation and analysis process. Some of these methods, as well as more advanced ones based on probabilistic inference networks, such as Bayesian networks, are implemented in analytical tools, such as Netica (https://www.norsys.com/).

Among the most advanced analytical tools is Cogent, a cognitive agent for intelligence analysis (Tecuci et al., 2015; 2018), which is the latest in a series of analytical tools that includes Disciple-LTA (Tecuci et al., 2007), TIACRITIS (Tecuci et al., 2011), and Disciple-CD (Tecuci et al., 2016a) An analyst collaborates with Cogent to answer an intelligence question by following a systematic approach grounded in the science of evidence (Schum, 2009) and the scientific method. First the analyst imagines possible answers to the addressed intelligence question in the form of competing hypotheses. Each hypothesis is then analyzed by developing a Wigmorean probabilistic inference network to assess its probability based on the available evidence (Wigmore, 1913). Evidence is any observable sign, datum, or item of information that is relevant in deciding whether a hypothesis is true or false (Schum, 2009). The hypothesis with the highest probability is proposed as the best answer to the intelligence question asked.

Building on Cogent, we proposed a logic and probability-based concept for how a cognitive agent for intelligence analysis could be connected to a real-time persistent processing system to enable automatic analysis. We developed a preliminary prototype system, called CAPIP (Cognitive Agent for Persistent Intelligence Processing), directly interacting with MITRE's Integrated Environment for Persistent Intelligence software (Tecuci et al. 2019). In this paper we present research that builds on CAPIP to develop a methodology and an instructable system for sensemaking in ISR. First we briefly introduce the Multi-Agent System for Sensemaking through Hypothesis generation and analysis (MASH), and summarize the methods it implements. Then we illustrate and discuss the experience with instructing MASH to make sense of complex situations.

2. Overview of MASH

The architecture of MASH is presented in Figure 1. MASH communicates with a set of ***Surveillance Agents*** developed to detect certain types of events, called alerts, such as the following one:

According to thermal imagery sensor using heat detection, several areas of the Destructville uranium enrichment plant are emitting heat as of 3/5/2020.

This is an alert because it may indicate a situation of interest, such as:

Shamland is producing centrifuge-enriched uranium at Destructville uranium enrichment plant as of 3/5/2020.

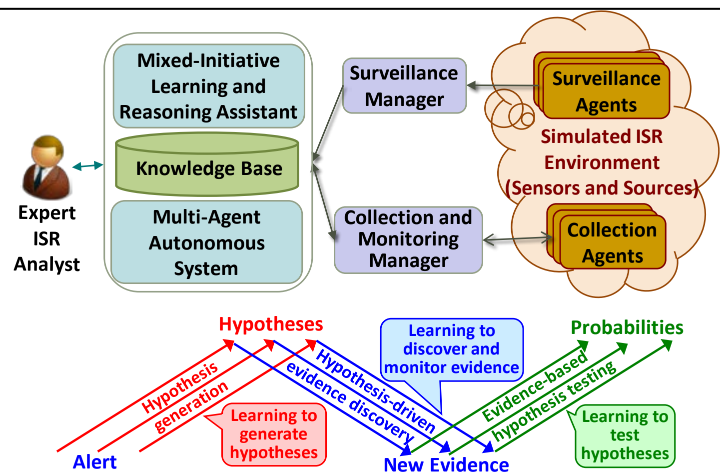

Figure 1. MASH, multi-agent system for sensemaking in ISR, learns to generate hypotheses from alerts, to discover evidence, and to test the hypotheses based on the discovered evidence.

Through *abductive reasoning* (which shows that something is *possibly* true) MASH generates other competing hypotheses that may explain the alert:

An explosion has occurred at the Destructville uranium enrichment plant as of 3/5/2020.

An attack against the Destructville uranium enrichment plant has occurred as of 3/5/2020.

To determine which of the competing hypotheses is true, MASH uses each hypothesis and *deductive reasoning (*which shows that something is *necessarily* true) to discover new evidence. It first identifies sufficient conditions or indicators that favor or disfavor the hypothesis and then, through the ***Collection and Monitoring Manager***, invokes specialized ***Collection Agents*** to look for evidence in the ISR environment.

Once additional evidence is discovered, MASH uses *inductive reasoning* (which shows that something is *probably* true) to test each hypothesis and determine whether *Shamland is producing centrifuge-enriched uranium*.

MASH is not connected to a real ISR environment. Instead we developed a ***Simulated ISR Environment*** that enables the testing of automatic sensemaking and also facilitates the transition to real data sources and real ISR environments.

Conventionally, a knowledge-based system like MASH is developed by a knowledge engineer who encodes the knowledge of an expert ISR analyst into the system's knowledge base. This is a difficult, time-consuming, and error-prone process, well-known as the *knowledge acquisition bottleneck* of the knowledge based systems development process (Feigenbaum, 1993; Buchanan and Wilkins, 1993; Tecuci et al., 2016a). Instead, MASH is an instructable system that is directly taught by an expert ISR analyst (with limited support from a knowledge engineer), in a way that is similar to teaching a student or collaborator, by demonstrating and explaining to the system how to generate and analyze hypotheses in a given situation. Successive versions of the instructional approach were presented in (Tecuci, 1998; Tecuci et al., 2000; 2002, 2005, 2008, 2016b). In the case of MASH, the expert ISR analyst demonstrates to the ***Mixed-Initiative Learning and Reasoning Assistant*** what hypotheses to generate from a specific alert, how to use each hypothesis to discover relevant evidence, and how to test each hypothesis based on the discovered evidence. Generalizing from the demonstrated reasoning example, MASH learns explicit *abductive rules* to generate competing hypotheses from alerts, *collection rules* to discover and collect relevant evidence, and *analysis rules* to test the hypotheses, as shown at the bottom of Figure 1. These rules are stored in the ***Knowledge Base***.

The ***Multi-Agent Autonomous System*** uses this knowledge base to automatically make sense of new situations. It includes several specialized autonomous agents, such as the ***Hypothesis Generation Agent***, which generates competing hypotheses from the alerts received from the ***Surveillance Manager,*** and the ***Hypothesis Analysis Agents,*** which analyze the competing hypotheses and generate evidence collection requests. The ***Collection and Monitoring Manager*** identifies which of the available Collection Agents can process a collection request, forwards it to those agents, and returns any discovered evidence to the corresponding Hypothesis Analysis Agent.

MASH can be instructed to perform a wide variety of simulated ISR tasks, such as:

- Automatically monitor plants and facilities on a global basis to determine construction or operating status, including production facilities as well as transportation infrastructure, such as pipelines, ports, and loading terminals.
- Automatically assess the intended purpose of facilities under construction worldwide as new information comes in and construction advances.
- Automatically assess indicators for determining the likelihood of whether an attack or explosion occurred at plants and facilities on global basis.

- Automatically monitor changes in security at industrial and military facilities on global basis.
- Automatically assess country capability to support production of various weapon systems.
- Automatically assess indicators of threat perceptions for multitude of countries.

In the next section we present MASH's hypothesis generation and analysis methods.

3. Hypothesis Generation and Analysis

3.1 Abduction-Based Hypothesis Generation

For thousands of years, the greatest human minds, including Aristotle (384BC–322BC), Galileo Galilei (1564–1642), Isaac Newton (1642–1727), John Locke (1632–1704), and William Whewell (1794–1866) considered that the only inferences governing hypothesis generation and analysis were *deduction* and *induction*. Only relatively recently Charles Peirce (1898; 1901) suggested that new ideas or hypotheses are generated through a different form of inference than deduction or induction, which he called *abduction* and associated with *imaginative reasoning*:

> *The surprising fact, C, is observed;*
> *But if A were true, C would be a matter of course,*
> *Hence, there is reason to suspect that A is true.*

Table 1 provides brief definitions of the deductive, inductive, and abductive inferences.

Table 1. Types of inferences.

Deductive Inference	$\forall x, U(x) \rightarrow V(x)$ $U(a_1)$ *Necessarily* $V(a_1)$	Whenever $U(x)$ is true, $V(x)$ is also true $U(a_1)$ is true Therefore $V(a_1)$ is *necessarily* true
Inductive Inference	$U(a_1)$ and $V(a_1)$ $U(a_2)$ and $V(a_2)$ … $U(a_n)$ and $V(a_n)$ $\forall x, U(x) \rightarrow$ *Probably* $V(x)$	When $U(a_1)$ was true, it was observed that $V(a_1)$ was also true When $U(a_2)$ was true, it was observed that $V(a_2)$ was also true … When $U(a_n)$ was true, it was observed that $V(a_n)$ was also true Therefore, whenever $U(x)$ is true, $V(x)$ is also *probably* true
Abductive Inference	$U(a_1) \rightarrow V(a_1)$ $V(a_1)$ *Possibly* $U(a_1)$	If $U(a_1)$ were true then $V(a_1)$ would follow as a matter of course $V(a_1)$ is true Therefore $U(a_1)$ is *possibly* true

Automatic hypothesis generation through abductive reasoning is computationally-intensive because there are numerous hypotheses that can be abduced from an observation (Josephson and Josephson, 1994; Schum, 2001a; Walton, 2005; Forbus, 2015; Langley, 2019).

We have developed an approach to hypothesis generation as a multi-step abductive process where each abductive step involving the generation of competing hypotheses is followed by evidence collection and testing of these hypotheses, to significantly reduce the hypothesis space. This approach is illustrated in Figure 2.

If we were to perform a single-step abduction, from evidence E to a hypothesis of interest that would explain it, we would obtain a huge number of hypotheses represented as dots at the top of the figure. We would then need to investigate each of these competing hypotheses to find the most

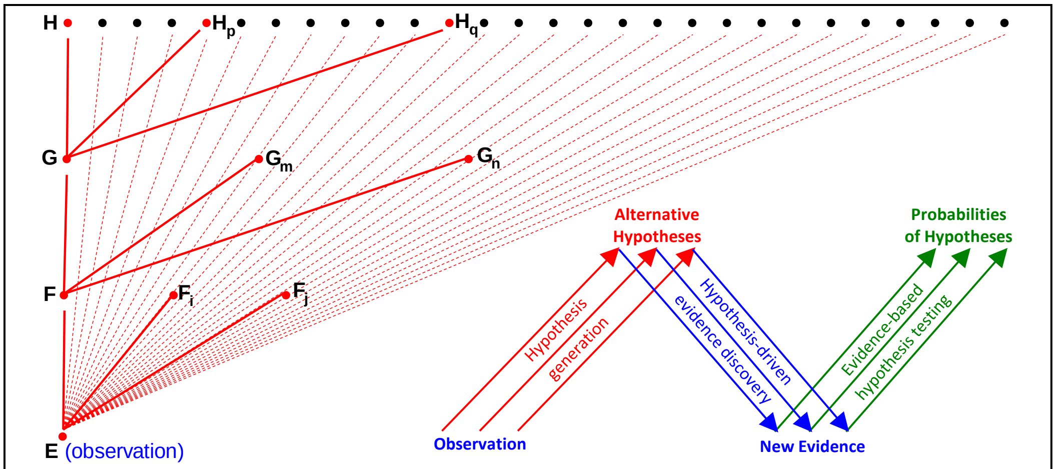

Figure 2. Multi-step abduction where each abduction of hypotheses is followed by deduction and induction to determine the most promising of the abduced hypotheses.

likely explanation. Now contrast this process with multi-step abduction. From E, one may abduce F, F_i, and F_j. At this point, we would search for evidence relevant to these three hypotheses and we would test them based on the discovered evidence concluding, for example, that F is the most likely. Then we would continue the abduction from F, abducing G, G_m, and G_n, testing these hypotheses, and concluding, for example, that G is the most promising. Finally, from G, we would abduce the hypotheses of interest H, H_p, and H_q, and test them. This approach to hypothesis generation based on spiral hybrid reasoning, where small abductive, deductive, and inductive steps feed each other, significantly reduces the hypothesis space.

Figure 3 illustrates the multi-step abduction process in the case of the Shamland situation discussed in Section 4.

3.2 Hypothesis-Driven Evidence Discovery

Evidence to test the generated hypotheses is discovered by using the hypotheses themselves. The question is: *What evidence would favor or disfavor hypothesis H?* H is decomposed into simpler hypotheses by considering both favoring arguments (supporting the truthfulness of H), under the left (green) square, and disfavoring arguments (supporting the falsehood of H), under the right (pink) square, as represented in Figure 4. Each argument is an independent strategy that shows that the H in question is true or false, and is characterized by a specific relevance or strength. The argument consists either of a single sub-hypothesis (e.g., H3) or a conjunction of sub-hypotheses (e.g., H1 & H2). The sub-hypotheses in an argument may represent necessary and sufficient conditions, sufficient conditions, or only indicators of the hypothesis above them. The sub-hypotheses from these arguments are further decomposed through other arguments, leading to simpler and simpler (sub-sub-)hypotheses until these (sub-sub-)hypotheses are simple enough to show what evidence may favor or disfavor them. At this point corresponding evidence collection agents are invoked .

Consider the following hypothesis from the Bogustan situation that will be discussed in Section 4:

Plumes have been observed at the Tanan chemical plant as of 2/25/2020.

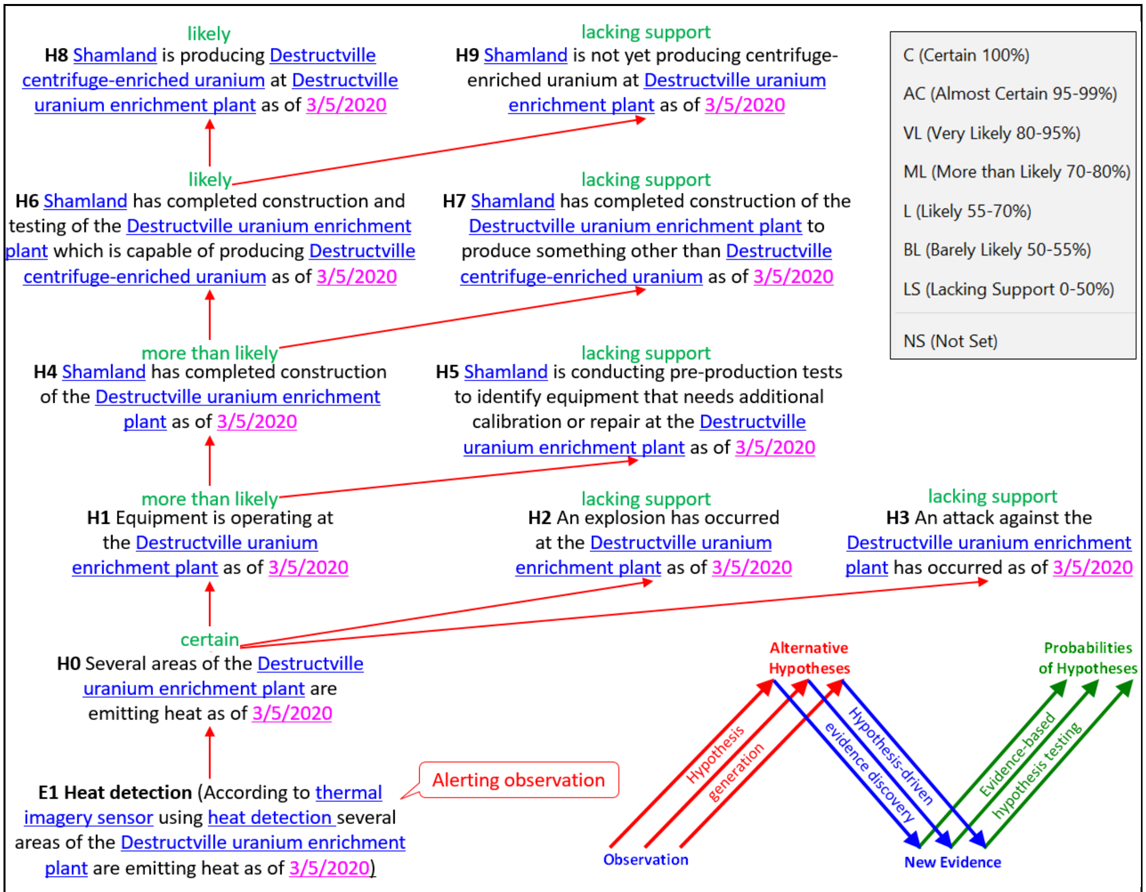

Figure 3. Multi-step abduction in the Shamland situation.
The generated hypotheses that lack evidentiary support are eliminated from the search space.

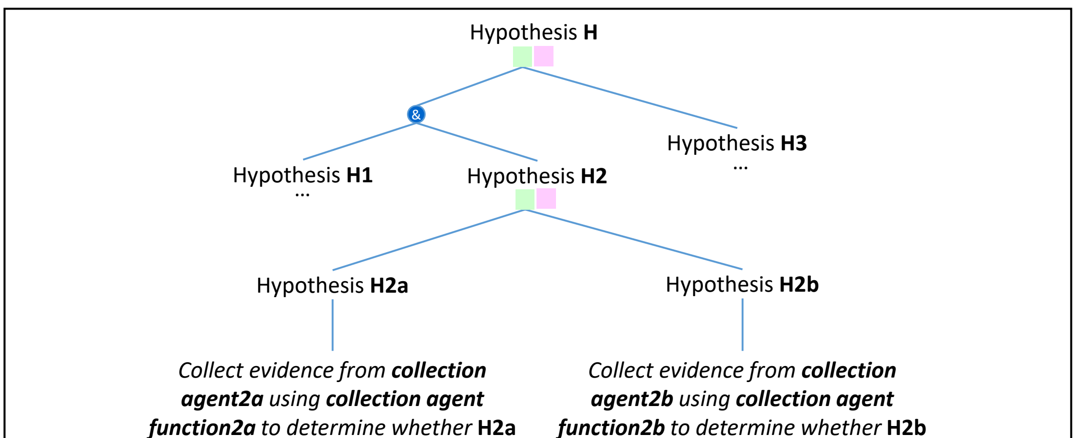

Figure 4. Hypothesis-driven evidence discovery: The top hypothesis is successively reduced to simpler and simpler sub-hypotheses and evidence is collected for the simplest sub-sub-hypotheses.

Evidence collection requests corresponding to this hypothesis include:

> Collect evidence from optical imagery sensor using plumes detection to determine whether plumes have been observed at the Tanan chemical plant as of 2/25/2020.
>
> Collect evidence from reporting source using collateral reporting to determine whether plumes have been observed at the Tanan chemical plant as of 2/25/2020.

3.3 Wigmorean Argumentation for Evidence-based Hypothesis Assessment

Once evidence relevant to a hypothesis is discovered, it can be used to test the hypothesis, as illustrated in Figure 5 that represents a Wigmorean argumentation. Wigmorean argumentations were initially introduced a century ago by Henry John Wigmore, a famous American jurist, as a graphical representation of how evidence supports or refutes claims in a court of law (Wigmore, 1913; 1937). They were resurrected by David Schum, who promoted their application both in law and in intelligence analysis (Schum, 1987; 2001b). Their logical structure was augmented with *Baconian probability* (Cohen, 1977; 1989) and *Fuzzy qualifiers* (Zadeh, 1983), such as 'likely' or 'almost certain' (Tecuci et al., 2016a, pp. 159-172). Consider, for example, sub-sub-hypothesis H2b. There are two items of evidence relevant to this hypothesis, the favoring evidence item E1, and the disfavoring evidence item E2. Each item of evidence has three credentials that need to be assessed: credibility, relevance, and inferential force. The *credibility* of evidence answers the question: "What is the probability that the evidence is true?" The *relevance* of evidence to a hypothesis answers the question: "What is the probability of the hypothesis being true if the evidence were true?" Based on these two credentials, MASH computes the *inferential force or weight* of the evidence on the hypothesis that answers the question: "What is the probability of the

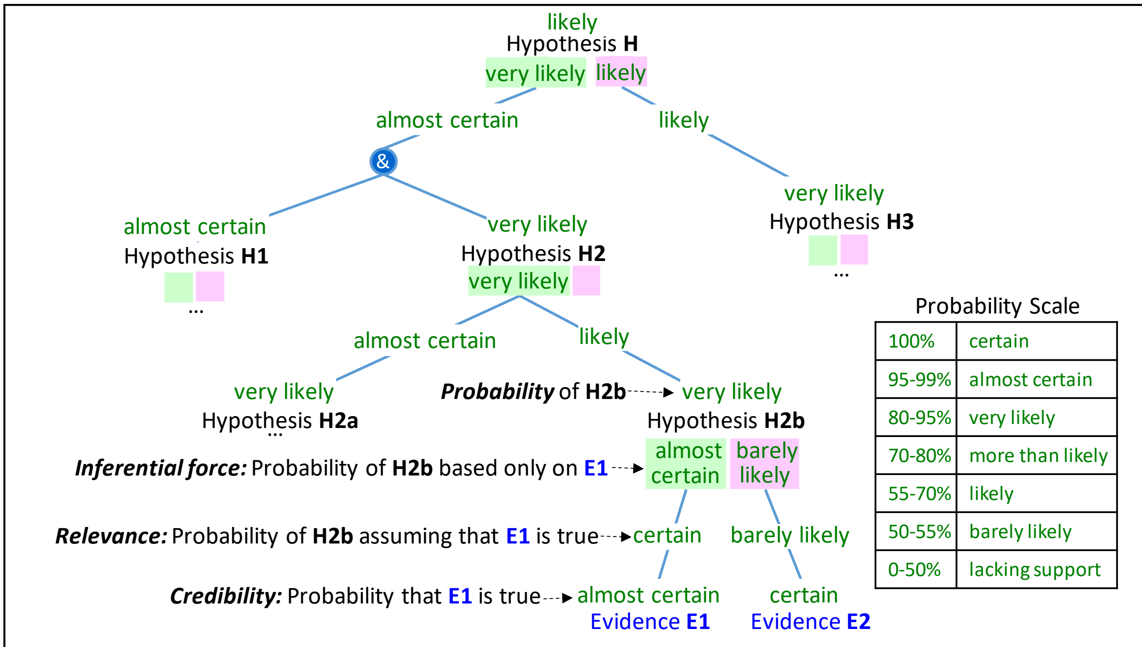

Figure 5. Wigmorean argumentation for evidence-based hypothesis assessment: The leaf hypotheses are assessed based on evidence and these assessments are combined to assess the top hypothesis.

hypothesis, based only on this evidence?" This is computed as the minimum between the credibility and relevance. For example, the inferential force of E1 is 'almost certain' and that of E2 is 'barely likely.'

The probability of sub-sub-hypothesis H2b is determined by balancing the inferential force of the favoring evidence with that of the disfavoring evidence. Once the probabilities of the bottom-level hypotheses have been determined based on evidence, the probabilities of the upper level hypotheses are computed based on the logical structure of the Wigmorean argumentation (conjunctions and disjunctions of hypotheses), using min-max probability combination rules common to the Fuzzy probability view (Zadeh, 1983; Negoita and Ralescu, 1975; Schum 2001b) and the Baconian probability view (Cohen, 1977; 1989; Schum, 2001b). These rules are much simpler than the Bayes rule used in the Bayesian probability view (Schum, 2001b), or the Dempster-Shafer rule used in the Belief Functions probability view (Shafer, 1976).

Such Wigmorean argumentations are easy to develop and understand, and an intelligent software system, such as MASH, can learn to generate them, as discussed in the next section.

4. Mixed-Initiative Teaching and Learning

4.1 Training Scenario

We will present the process of instructing MASH to automatically detect when a country is producing weapons of mass destruction at a certain plant. We will start with the following specific situation, referred to as "Bogustan:"

- *The country Bogustan was building a new chemical plant at Tanan that was nearing completion; the plant's purpose was not known.*
- *Bogustan was suspected of harboring weapons of mass destruction ambitions.*
- *A reconnaissance asset conducting a routine quarterly overflight detected heat signatures at the Tanan facility in late February.*

The expert analyst will demonstrate the analysis to answer the question

Is Bogustan producing chemical warfare agents at the Tanan chemical plant as of 2/25/2020?

and MASH will learn general rules for answering questions of the type:

Is country producing weapons of mass destruction at plant as of date?

While developing the analysis, the expert analyst (with support from a knowledge engineer) will also extend the agent's ontology with the instances and concepts used in the analysis.

The expert analyst will then present MASH with the "Shamland" situation:

- *The country Shamland was building a large plant at Destructville, whose purpose was not known.*
- *Shamland was suspected of wanting to develop nuclear weapons.*
- *A reconnaissance asset conducting a routine quarterly overflight detected heat at the Destructville facility in early March.*

MASH will automatically generate hypotheses, collect evidence, and test the hypotheses to answer the question:

Is Shamland producing Destructville centrifuge-enriched uranium at the Destructville uranium enrichment plant as of 3/5/2020?

The expert analyst will then check the analysis generated by MASH, correct the mistakes made in the argumentation (if any), and complete the analysis (if necessary).

4.2 Expert-Provided Demonstration of Sensemaking

MASH does not start with an empty knowledge base. It starts with a knowledge base containing general evidence-based reasoning (EBR) knowledge, including an ontology of evidence and general rules for assessing the credibility of evidence. It also starts with general ISR knowledge in the form of an ontology of sensors that are useful in any ISR application.

The ontology language is an extension of RDFS (W3C, 2004) with additional features for learning and evidence representation. The rules are IF-THEN structures with first-order logic applicability conditions expressed using the concepts from the ontology (Tecuci et al. 2016b).

Table 2 summarizes this initial ontological knowledge.

Table 2. Knowledge base before agent instruction.

Evidence-based reasoning				ISR		
Instances	Concepts	Features	Rules	Instances	Concepts	Features
17	72	76	7	7	6	1

Figure 6 shows the top-level of the analysis demonstrated by the expert ISR analyst who builds it by using the modeling editor of MASH. Table 3 summarizes the corresponding knowledge elements in this analysis and in the knowledge base.

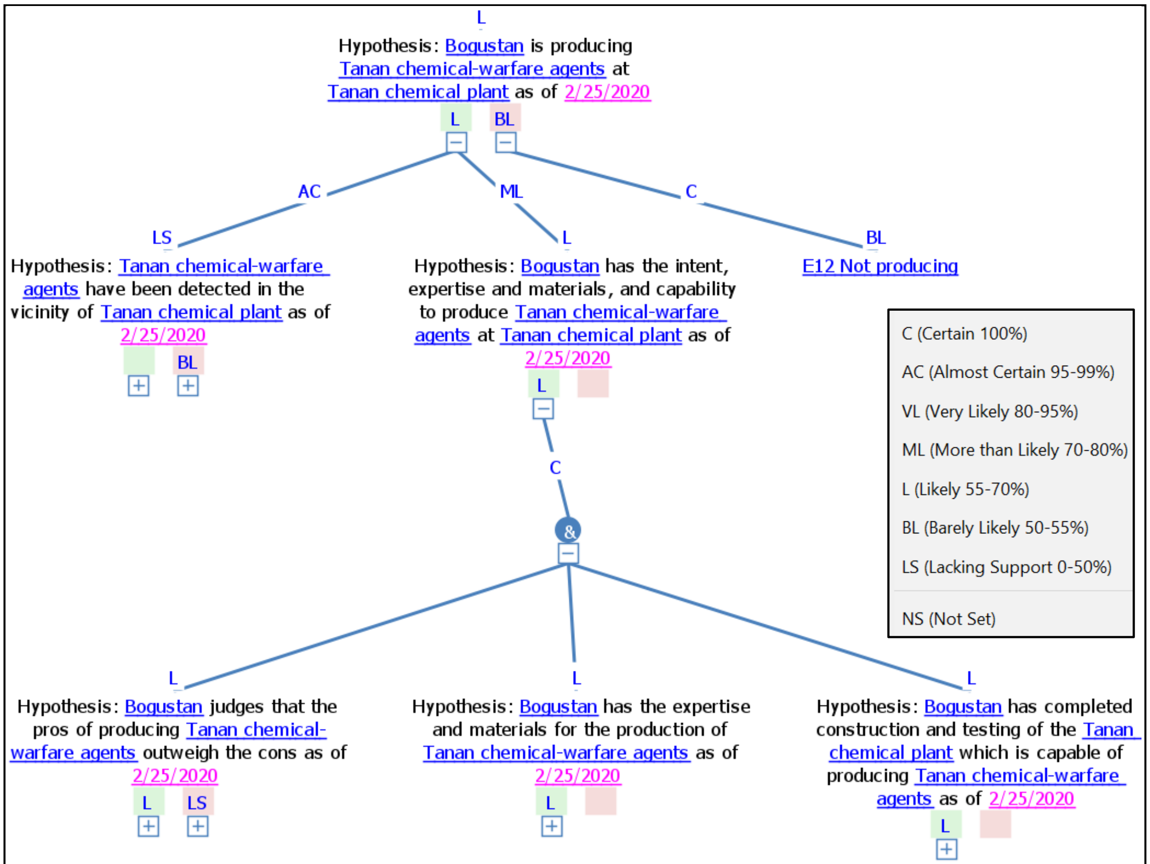

Figure 6. The top hypothesis has two favoring sub-hypotheses under the left (green) square, and a disfavoring evidence under the right (pink) square. The second sub-hypothesis has three indicators.

Table 3. Knowledge elements used in the demonstrated analysis.

Hypotheses	Reductions	Evidence items	Scenario instances	Domain concepts	Domain features
44	24	29	9	26	6

The left-hand side of Figure 7 shows a fragment of the ontological description of the Bogustan situation that includes information such as:

Bogustan has Halifaza as enemy.

Tanan chemical plant belongs to Bogustan and may produce chemical warfare agents.

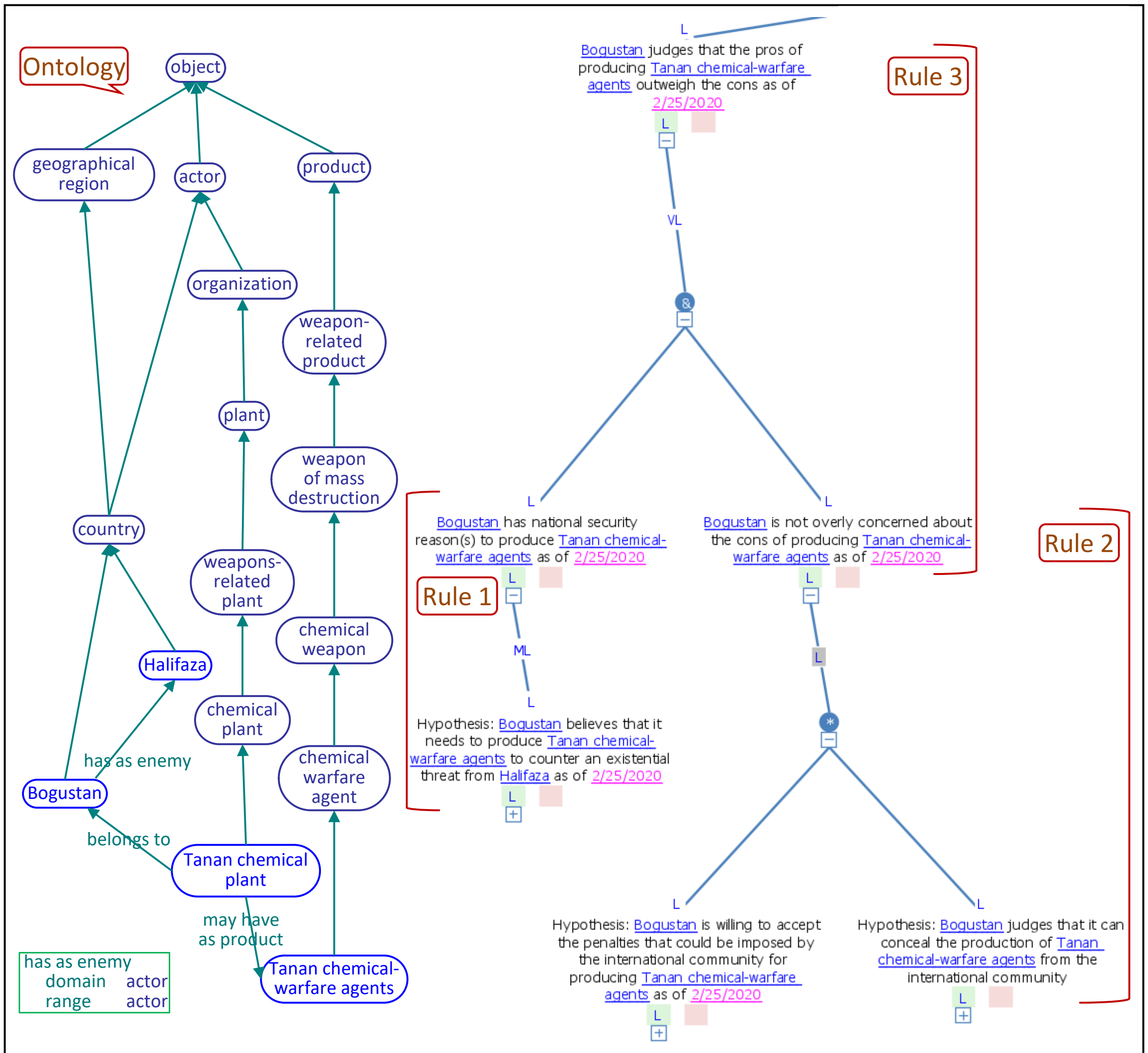

Figure 7. Ontology-based rule learning: MASH uses the ontology in the left-hand side to learn the three rules in the right-hand side by generalizing the hypotheses reduction provided by the expert.

As shown in the right hand side of Figure 7, from each demonstrated argument, MASH learns general analysis rule. In particular, it learns Rule 1, Rule 2, and Rule 3 as ontology-based generalizations of the demonstrated arguments. The next sections discuss the rule learning process.

4.3 Rule Learning

Figure 8 illustrates the learning of Rule 1 from the argument summarized as:

> *Bogustan has national security reasons to produce chemical warfare agents because it believes it needs them to counter an existential threat from Halifaza.*

The expert ISR analyst browses the ontology and selects the fragment that justifies the argument, that is: Bogustan has as enemy Halifaza . We call this an *explanation* of the argument. Using it, MASH automatically generates the analysis rule from the right-hand side of Figure 8, as follows:

- It generates a general argument by replacing each instance in the example argument with a different variable (i.e., Bogustan with ?O1, Tanan chemical warfare agents with ?O2, Halifaza with ?O3, and 2/25/2020 with ?S1).
- It uses the ontology to compute the applicability condition of the general argument that shows the possible values that these variables may take.

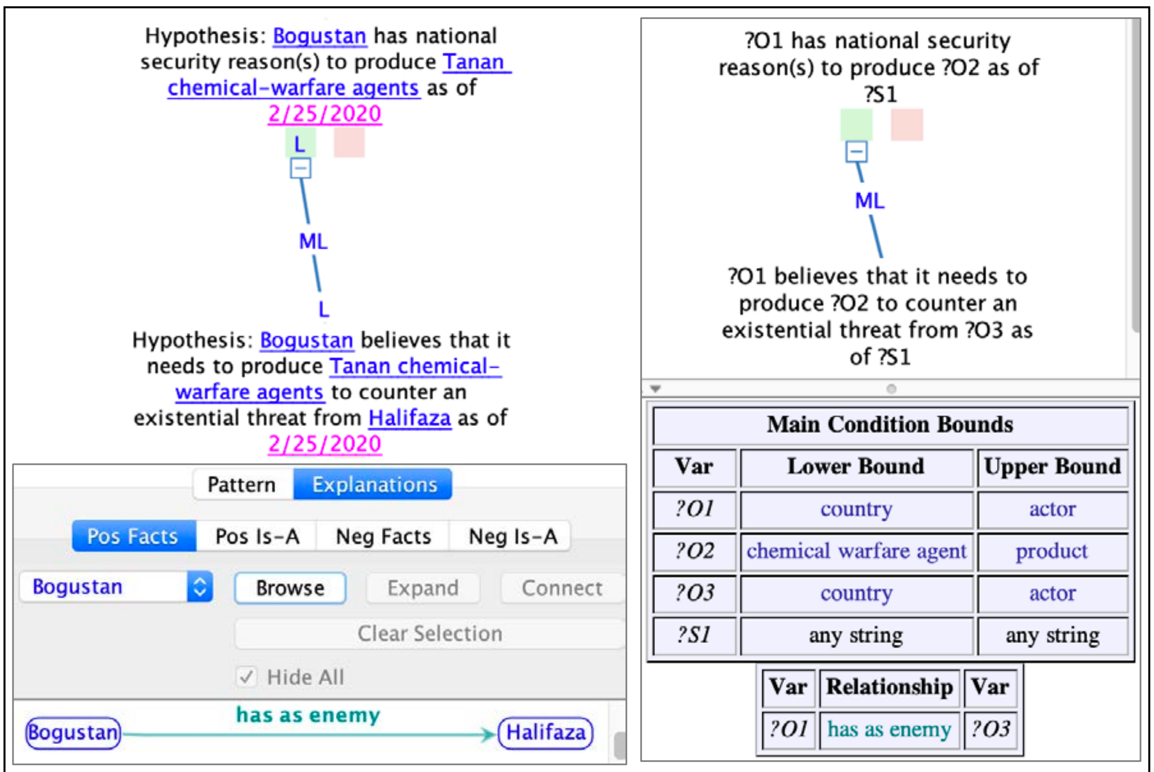

Figure 8. Mixed-initiative rule learning: The ISR analyst provides the hypothesis reduction and selects the ontology fragment from the left side as explanation; MASH generates the rule from the right-hand side.

Notice however that, instead of a single applicability condition, MASH learns a lower bound for this condition and an upper bound, by using two complementary learning strategies:

- The upper bound of the condition is generated by employing the strategy of an *agressive learner* that wants to maximize the opportunities of employing the learned rule. For example, the upper bound of ?O1 is obtained as the maximal generalization of Bogustan. According to the generalization hierarchy represented by the ontology in Figure 7, the maximum generalization of Bogustan is object. However, Bogustan is also the value of the feature "has an enemy" whose range is actor, a sub-concept of object. Therefore, ?O1 in the upper bound of Rule 1 can be instantiated by any instance of actor. This strategy increases the number of situations where Rule 1 can be applied, but in some of these situations the reasoning may not be correct.
- The lower bound of the condition is generated by employing the strategy of a *cautious learner* that wants to minimize the chances of making mistakes when employing the learned rule. In this case the lower bound of ?O1 is obtained as the minimal generalization of Bogustan, which is country. This strategy increases the confidence of the agent in the correctness of its reasoning, but the agent may fail to apply the reasoning rule in situations where, in fact, it is applicable.

The two bounds may be refined, and may even become identical, based on additional examples encountered by the agent and their explanations.

Rule 2 and Rule 3 are learned in a similar way, both based on the following explanations:

Tanan chemical plant belongs to Bogustan
Tanan chemical plant may have as product Tanan chemical warfare agents

The vast majority of the current machine learning approaches are heavily statistical and learn single functions from a large number of examples. Such approaches are not applicable for learning to generate and analyze hypotheses in the ISR domain because such sets of examples do not exist and would be very difficult to create. Instead, the expert analyst just shows the agent how s/he analyzes the current situation and the agent learns rules as ontology-based generalizations of the demonstrated reasoning steps, as discussed above. The explanations that point directly to the relevant features of the individual examples enable rapid learning. Thus, these features do not need to be discovered through the statistical comparison of a large number of positive and negative examples (that are not available anyway), as done in the inductive learning methods (Witten et al. 2011; Flach, 2012; Alpaydyn, 2020).

4.4 Automatic Analysis

From the demonstrated analysis, a fragment of which was shown in Figure 6, MASH learned 24 analysis rules. These rules enabled the system to automatically generate the analysis of the following hypothesis:

Shamland is producing Destructville centrifuge-enriched uranium
at Destructville uranium enrichment plant as of 3/5/2020.

Figure 9 shows a fragment of the generated analysis. While most of this reasoning is correct, the analyst judges that the instance of Rule 1 is incomplete, and will interact with the agent to refine it as explained next.

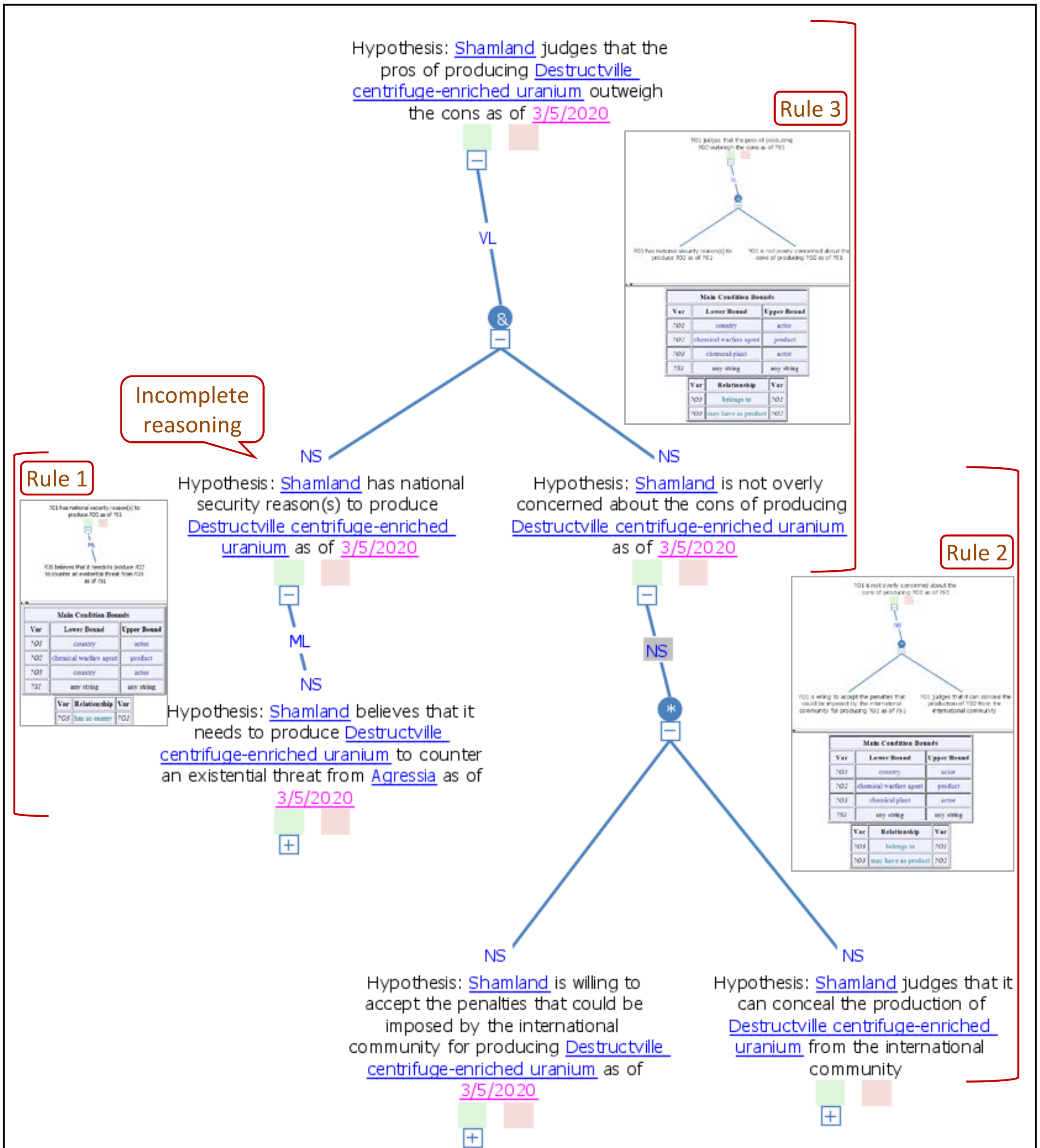

Figure 9. Automatically generated analysis: MASH generated this analysis fragment by applying the rules learned from the analysis fragment shown in Figure 7.

4.5 Rule Refinement with a Negative Example

The incomplete reasoning from Figure 9 is shown again in the middle of Figure 10. The analyst considers that this reasoning step is incomplete because it should contain an additional sub-hypothesis:

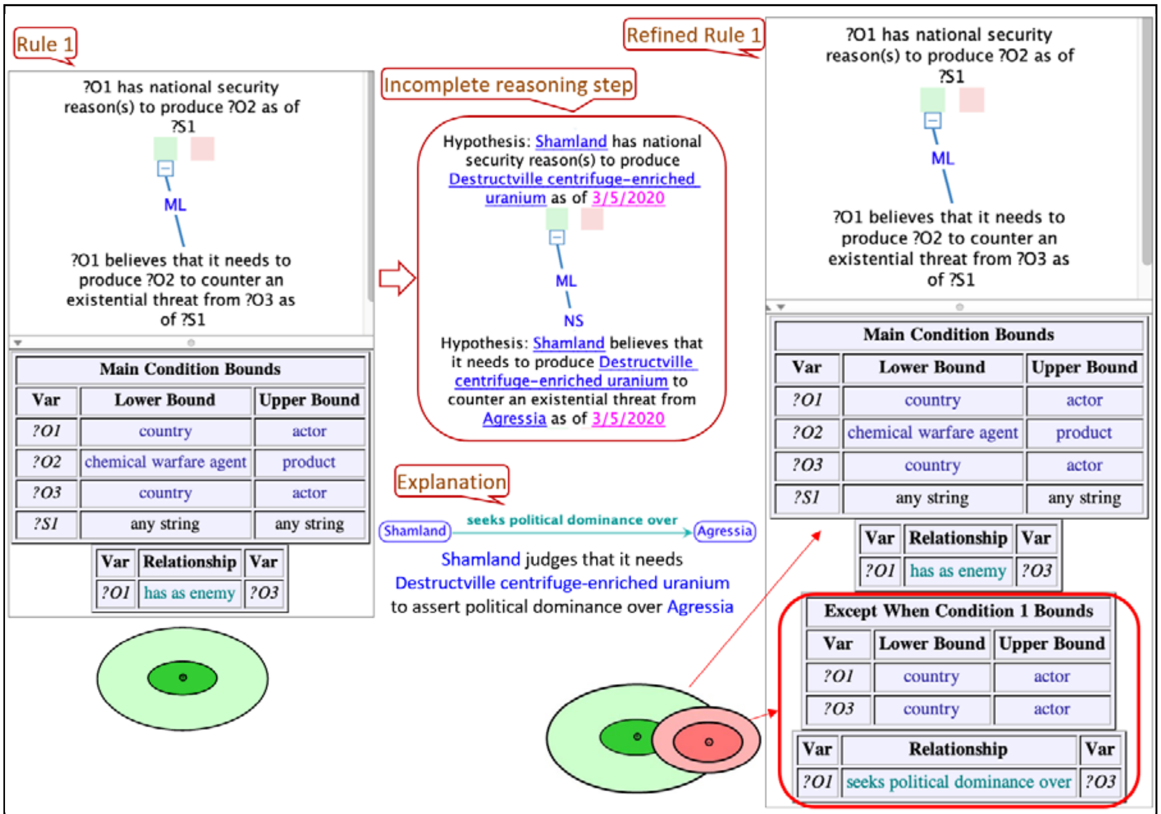

Figure 10. Rule refinement based on a negative example: The ISR expert indicates that the instance of Rule 1 from the middle of the figure is incomplete because of the ontological relationship shown below it. As a result, MASH specializes Rule 1 by adding an Except When condition, to no longer generate the incomplete reasoning step.

Shamland judges that it needs Destructville centrifuge-enriched uranium to assert political dominance over Agressia.

The corresponding explanation for the omitted sub-hypothesis is: Shamland seeks political dominance over Agressia

As a result, MASH refines Rule 1 by adding an *Except-When* applicability condition, as shown in the right-hand side of Figure 10. Now Rule 1 is applicable if the *Main Condition* is satisfied and the *Except-When Condition* is not satisfied. Therefore it no longer applies to the Shamland situation.

Thus, through a few simple expert-agent interactions, the agent learns a complex reasoning rule. In conventional agent development approaches, such rules have to be manually defined by a knowledge engineer.

After Rule 1 is refined, the expert extends the incomplete reasoning step with the additional sub-hypothesis and the agent learns a new rule for situations where an actor seeks political dominance over another actor.

Of the 24 analysis rules learned from the Bogustan situations, 20 were correctly applied in the Shamland situations, 4 (including Rule 1) were incorrectly applied, and 4 new rules were learned.

5. Conclusions

The presented approach to automated analysis in ISR has several very significant advantages over the current manual approach in terms of speed, quality, and transparency of analysis. Over time, as the knowledge base is developed and refined for different situations, the gain in terms of speed, quality, and consistency will grow at an increasing rate.

As summarized below, the extra effort needed to train the agent is very limited when compared to the advantages of automated ISR and the significant expected reduction in the overall workload of the ISR analysts.

Automation of ISR Analysis

Hypothesis generation and analysis in ISR are not currently automated. These tasks have to be performed by the ISR analyst repeatedly for each new situation.

This same ISR analyst instructs MASH how to reason by developing the analysis for a given situation. In this case, however, MASH also supports the analyst in developing a more comprehensive, defensible, and persuasive analysis by following the approach introduced in Section 3.

Thus, the ISR analyst needs to manually perform the original analysis to answer the relevant question but when using MASH, the analyst is developing a more complete and systematic analysis that is also used to train the system. Our approach to learning analytic rules from this example analysis also reduces this effort as compared with having a knowledge engineer manually define the rules, as is the case in a conventional agent development approach. Moreover, the additional effort for instructing MASH is dwarfed by the effort saved in performing future analyses which are entirely or at least partially generated automatically.

Early and Continuous System Use Due to Incremental Training

MASH does not need to be fully trained before being used. We are developing a flexible control structure, enabling the system to operate with different levels of autonomy.

At the beginning, MASH may operate under the full control of the analyst, who will not only use it to make sense of a situation, but also to train it based on the developed analysis.

As MASH learns, the control changes to a mixed-initiative one, where some parts of the analysis are generated by MASH and the entirely new parts are developed by the analyst. For example, certain basic components of many analytic problems, such as intent, could be automatically generated and then modified to take into account unique differences related to the decision maker. Thus, analysis of information related to a particular problem helps to inform analysis in other related problems. Analysis in this manner systematically builds upon itself. A complete set of potential indicators can be developed for a wide range of intelligence questions, such as the operating status of industrial facilities or the intended purpose of a plant under construction.

MASH continuously learns from the analyst, and its contribution to the analysis process continuously increases, enabling it to operate autonomously in more and more situations.

Automated Hypothesis Generation, Evidence Collection, and Hypothesis Testing

MASH automatically generates a comprehensive and transparent analysis by producing a schematic diagram that completely lays out the underlying analytic framework for every hypothesis. This includes the connection between the evidence and various intermediate conclusions in the analysis, the evaluation of the credibility of evidence and its strength in supporting a conclusion, and the role of any assumptions in addressing missing information.

MASH will almost instantaneously generate the hypotheses explaining an ambiguous alert and their argumentation structures, incomparably faster than the current manual process, due to:

- Synergistic integration of abduction, deduction, and induction operations that feed each other, as discussed in Section 3.1;
- Use of learned hypothesis analysis rules with applicability conditions.

The overall duration of hypothesis testing will depend almost entirely on the speed with which the sensors/information sources return the requested information. But even this process is much faster because, instead of searching through a huge amount of information in the hope of finding something useful, as is done in the current manual approach, MASH uses the hypotheses themselves to guide the evidence search, and very rapidly discover the useful evidence, as discussed in Section 3.2.

Rapid Adaptation to Changes in the Situation

A significant challenge of ISR is that sensemaking must be made in a continuously changing world. An explanation for some pattern of past events previously regarded as correct may now become less persuasive in light of new evidence just discovered today. A prediction regarded as highly likely yesterday may be overtaken by today's events. In fact, the very questions asked yesterday may need to be revised or may even seem unimportant in light of what is learned today.

MASH continuously monitors and automatically detects new or changed evidence, and rapidly regenerates the analysis. Thus, new developments are processed and assessed quickly, reducing the chances of surprise.

More Fruitful Coordination and Focused Analytic Review

MASH, with its explicit argumentation and explicitly justified assessments, will enable analysts to zero in on where in the analysis they agree and disagree. As a result, less time will be spent on areas where analysts agree, and more time can be spent on more fruitful discussions of analytic differences. New information can be more efficiently discussed within this existing framework of agreement and disagreement.

The system significantly improves the transparency of the analytic process. The automated analysis builds an explicit and intuitive argumentation that shows very clearly how the conclusions emerge from evidence. The argumentation also includes justifications for any assumptions made, the assessments related to the credibility of the evidence, and the relevance of the evidence and sub-hypotheses to higher-level hypotheses. Managers can use the systematized argumentation that illustrates the analytic path to the main judgments in their review of analytic products.

The automated analysis allows for real-time adjustments to various assumptions to determine their impact on the analysis' main judgment. This capability, in turn, provides a basis for analysts to assess their confidence level in the main judgment. For example, analysts' confidence in judgments that vary significantly with only small changes in their assumptions would be lower than their confidence in judgments that vary only slightly. Moreover, the automated, systematized analysis will allow analysts to highlight in their finished reports those assumptions that are critical to their main judgments.

More Systematic Record Keeping and On-the-Job Analyst Training

The automated system will create an archive of analyses that will facilitate the training of new analysts as well as facilitate post-mortem analyses of controversial analytic judgments. Analyst training on specific issues will be less ad hoc and less dependent on the specific tendencies of individual analysts. The automated system creates an extensive and complete record of the basis for various analytic judgments that will allow mentors to spend less time on one-on-one training for newly assigned analysts. Analysts assigned to new teams can first review this database of analyses to understand (as well as question) how the team's analytic judgments were reached. This will allow new analysts to more quickly assimilate the basis for the teams' analyses and to allow more fruitful and focused discussions between mentors and new analysts.

In cases where the analysis turns out to be wrong, the automated system can provide a specific blueprint to facilitate the identification of where in the analysis faulty assumptions and sub-judgments were made, or the credibility of information was not assessed properly. Lessons learned through such reviews can be quickly and systematically transferred to existing knowledge bases to address similar issues.

References

Alpaydyn, A. (2020). *Introduction to Machine Learning*. MIT Press.

Boicu, M., Tecuci, G., Stanescu, B., Marcu D., and Cristina E. Cascaval, C. E. (2001). Automatic Knowledge Acquisition from Subject Matter Experts, in P*roceedings 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.

Buchanan, B. G. and Wilkins, D. C., eds. (1993). *Readings in Knowledge Acquisition and Learning: Automating the Construction and Improvement of Expert Systems*, Morgan Kaufmann, San Mateo, CA.

Cohen, L. J. (1977). *The probable and the provable*. Oxford, UK: Clarendon Press.

Cohen, L. J. (1989). *An introduction to the philosophy of induction and probability*. Oxford, UK: Clarendon Press.

DOD J. P. 1-02 (2010). Department of Defense Dictionary of Military and Associated Terms, *Joint Publication 1-02*, 8 November 2010.

Feigenbaum, E. A. (1993). *Tiger in a Cage: The Applications of Knowledge-Based Systems*, Invited Talk, AAAI-93 Proceedings, http://www.aaai.org/Papers/AAAI/1993/AAAI93-127.pdf

Flach, P. (2012). *Machine Learning: The Art and Science of Algorithms that Make Sense of Data*. Cambridge University Press.

Forbus, K. (2015). Analogical abduction and prediction, *2015 AAAI Fall Symposium - Technical Report*.

Heuer R. J. Jr., and Pherson R. H. (2011). *Structured Analytic Techniques for Intelligence Analysis*, CQ Press.

Josephson, J. R., and Josephson, S. G. (1994). *Abductive Inference: Computation, Philosophy, Technology*. Cambridge University Press.

Langley, P. (2019). Heuristic Construction of Explanations Through Associative Abduction, *Proceedings of the Seventh Annual Conference on Advances in Cognitive Systems*, Technical Report Number COLAB2-TR-4, 21-36, Massachusetts Institute of Technology, Cambridge, Massachusetts, August 2-5.

Moore, D. T. (2001). *Sensemaking: A Structure for Intelligence Revolution*. NDIC Press.

Negoita, C. V., and Ralescu, D. A. (1975). *Applications of Fuzzy Sets to Systems Analysis*. Wiley: New York, New York.

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

Peirce, C. S. (1901). Abduction and Induction, in Buchler, J. (ed.) (1995). *Philosophical Writings of Peirce*, 150-156, Dover: New York, NewYork.

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

Schum, D. A. (2001a). Species of abductive reasoning in fact investigation in law. *Cardozo Law Review*, *22*, 1645–1681.

Schum, D. A. (2001b). *The evidential foundations of probabilistic reasoning*. Evanston, IL: Northwestern University Press.

Schum, D. A. (2009). Science of Evidence: Contributions from Law and Probability. Law Probab Risk 8 197–231.Shafer, G. (1976). *A Mathematical Theory of Evidence*. Princeton University Press: Princeton, New Jersey.

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

Tecuci G., Boicu M., Bowman M., Marcu D., Shyr P., Cascaval C. E. (2000). An Experiment in Agent Teaching by Subject Matter Experts, *International Journal of Human-Computer Studies*, pp. 583-610, 53.

Tecuci, G., Boicu, M., Marcu, D., Stanescu, B., Boicu, C., Comello, J., Lopez, A., Donlon, J., Cleckner, W. (2002) Development and Deployment of a Disciple Agent for Center of Gravity Analysis, in *Proceedings of the Eighteenth National Conference of Artificial Intelligence and the Fourteenth Conference on Innovative Applications of Artificial Intelligence*, AAAI-02/IAAI-02, pp. 853-860, Edmonton, Alberta, Canada, AAAI Press/The MIT Press.

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., Marcu, D., Boicu, C., Barbulescu, M., Ayers, C., Cammons, D. (2007). Cognitive Assistants for Analysts, *Journal of Intelligence Community Research and Development (JICRD)*, 2007. Also published in John Auger, William Wimbish (eds.), *Proteus Futures Digest: A Compilation of Selected Works Derived from the 2006 Proteus Workshop*, Joint publication of the National Intelligence University, Office of the Director of National Intelligence, and U.S. Army War College Center for Strategic Leadership, 2007, pp. 303-329. http://lac.gmu.edu/publications/2007/TecuciG_Cognitive_Assistants.pdf

Tecuci, G., Boicu, M., Marcu, D., Boicu, C., Barbulescu, M. (2008). Disciple-LTA: Learning, Tutoring and Analytic Assistance, *Journal of Intelligence Community Research and Development* (JICRD), July.

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.

Tecuci, G., Marcu, D., Boicu, M., Schum, D. A. (2015). COGENT: Cognitive Agent for Cogent Analysis, in *Proceedings of the 2015 AAAI Fall Symposium "Cognitive Assistance in Government and Public Sector Applications"*, Arlington, VA, 12-14 November.

Tecuci, G., Schum, D. A., Marcu, D., Boicu, M. (2016a). *Intelligence analysis as discovery of evidence, hypotheses, and arguments: Connecting the dots*. New York, NY: Cambridge University Press.

Tecuci, G., Marcu, D., Boicu, M., Schum, D. A. (2016b). *Knowledge engineering: Building cognitive assistants for evidence-based reasoning*. New York, NY: Cambridge University Press.

Tecuci G., Kaiser L., Marcu D., Uttamsingh C., Boicu M. (2018). Evidence-based Reasoning in Intelligence Analysis: Structured Methodology and System, *Computing in Science and Engineering*, 20(6) 9-21, November/December.

Tecuci G., Meckl S., Marcu D., Boicu M. (2019). Instructable Cognitive Agents for Autonomous Evidence-Based Reasoning, *Advances in Cognitive Systems*, Vol. 8, 2019. Also in *Proceedings of the Seventh Annual Conference on Advances in Cognitive Systems*, Technical Report Number COLAB^2-TR-4, pp.183-204, August 2-5, 2019, Massachusetts Institute of Technology, Cambridge, MA.

Walton, D. (2005). *Abductive Reasoning*. The University of Alabama Press.

Wigmore, J. H. (1913). The problem of proof. *Illinois Law Review*, *8*, 77–103.

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

Witten, I., Frank, E., and Hall, M. (2011). *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann.

W3C (2015). Semantic Web, http://www.w3.org/standards/semanticweb/

Zadeh, L. (1983). The role of fuzzy logic in the management of uncertainty in expert systems. *Fuzzy Sets and Systems*, *11*, 199–227.