

AFRL-IF-RS-TR-2007-174
Final Technical Report
September 2007



I2AT: THE INFORMATION AND INTERPRETATION ANALYSIS TOOLKIT

BAE Systems

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

STINFO COPY

**AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK**

NOTICE AND SIGNATURE PAGE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

This report was cleared for public release by the Air Force Research Laboratory Public Affairs Office and is available to the general public, including foreign nationals. Copies may be obtained from the Defense Technical Information Center (DTIC) (<http://www.dtic.mil>).

AFRL-IF-RS-TR-2007-174 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

FOR THE DIRECTOR:

/s/

JEFFREY W. HUDACK
Work Unit Manager

/s/

JOSEPH CAMERA, Chief
Information & Intelligence Exploitation Division
Information Directorate

This report is published in the interest of scientific and technical information exchange, and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

REPORT DOCUMENTATION PAGE				<i>Form Approved</i> OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Washington Headquarters Service, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington, DC 20503.					
PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.					
1. REPORT DATE (<i>DD-MM-YYYY</i>) SEP 2007		2. REPORT TYPE Final		3. DATES COVERED (<i>From - To</i>) Jun 05 – May 07	
4. TITLE AND SUBTITLE I2AT: THE INFORMATION AND INTERPRETATION ANALYSIS TOOLKIT				5a. CONTRACT NUMBER FA8750-05-C-0221	
				5b. GRANT NUMBER 	
				5c. PROGRAM ELEMENT NUMBER 62702F	
6. AUTHOR(S) Dan Hunter, James Melhuish, Andy Seidel, Jorge Tierno				5d. PROJECT NUMBER 459E	
				5e. TASK NUMBER B6	
				5f. WORK UNIT NUMBER 01	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) BAE Systems Advanced Information 6 New England Executive Park Burlington MA 01803-5012				8. PERFORMING ORGANIZATION REPORT NUMBER 	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) AFRL/IFED 525 Brooks Rd Rome NY 13441-4505				10. SPONSOR/MONITOR'S ACRONYM(S) 	
				11. SPONSORING/MONITORING AGENCY REPORT NUMBER AFRL-IF-RS-TR-2007-174	
12. DISTRIBUTION AVAILABILITY STATEMENT APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED. PA# AFRL-07-0086					
13. SUPPLEMENTARY NOTES 					
14. ABSTRACT The I2AT software uses Bayesian Networks, a probabilistic modeling framework, augmented with a suite of algorithms for analyzing hypotheses, data, and value of additional information. Automated generation of potential interpretations will help reduce the time needed to assimilate and act on new information. At the same time, flagging new data that is inconsistent with existing information will identify potential errors in the knowledge acquisition process or adversary attempts at deception, enabling preemptive correction before erroneous interpretations precipitate actions. I2AT implements an error model for information sources into the Bayesian Network that allows it to reason about the correctness of particular reports and what new evidence would best resolve ambiguities.					
15. SUBJECT TERMS Bayes, Bayesian model, hypothesis, validation, probability, information value, contradiction, evidence, error model, knowledge acquisition, deception					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT UL	18. NUMBER OF PAGES 58	19a. NAME OF RESPONSIBLE PERSON Jeffrey W. Hudack
a. REPORT U	b. ABSTRACT U	c. THIS PAGE U			19b. TELEPHONE NUMBER (<i>Include area code</i>) N/A

TABLE OF CONTENTS

<u>Section</u>	<u>Page</u>
List of Figures	iii
List of Tables	iv
1 Executive Summary	1
2 Introduction.....	2
2.1 the Information Interpretation Problem.....	2
2.2 Technical Challenges.....	2
2.3 The I2AT Solution.....	4
3 Modeling in I2AT	5
4 I2AT Approach	10
4.1 Data Conflict.....	10
4.2 Data Confidence	11
4.3 Hypothesis Confidence.....	12
4.4 Value of Information	13
4.5 Weight of Evidence	15
4.6 Error Modeling	15
5 Previous Work in Data Validation.....	20
6 Learning Error Models.....	21
6.1 Learning Framework	21
6.2 Results.....	24
7 I2AT Interface.....	26
8 Reasoning across Scenarios	30
9 Applying I2AT to Web-based Data Validation	34
9.1 Open Source Information Gathering	34
9.2 Web-based Data Validation.....	35
9.3 Limitations of Dr. Knowledge.....	39
10 Modeling Collaborative Intelligence Analysis	39
10.1 WMD Example.....	39
10.2 Threat of War Example	44
11 Extensions to I2AT	47
11.1 Causal Reasoning with JCAT.....	47
11.2 Joint Data/Model Validation	48
11.3 Time-Dependent Validation Schemes.....	49
11.4 Open Source Information Gathering	49
12 References.....	50
13 Appendix: model constraints	50

13.1Node Type Constraints50

13.2Metadata Requirements51

13.3Structural Requirements51

LIST OF FIGURES

Figure 1. I2AT architecture.....	4
Figure 2. Bayes net fragment for pilots' reports.....	6
Figure 3. Bayes net fragment for ELNOT sensor.....	7
Figure 4. Bayes net fragment for stationary rotor sensor.....	8
Figure 5. Entire Bayes net for BDA model.....	8
Figure 6. Full States and Probabilities for Bayes net BDA model	9
Figure 7. BDA model with BDA reports entered as evidence.....	11
Figure 8. Initial evidence for "Radar Destroyed" hypothesis.	12
Figure 9. Probability of hypothesis changes with new evidence.	12
Figure 10. Beta pdf for $a = 15$, $b = 10$	18
Figure 11. Beta pdf for $a = 6$, $b = 4$	18
Figure 12. Distribution of ages of terrorists.....	20
Figure 13. The Learning Process.	22
Figure 14. Testing (and Scoring) the Learned Model.....	23
Figure 15. Model for Testing Data Validation Framework.	24
Figure 16. Performance of Voting EM Algorithm on Test Model.	25
Figure 17. I2AT User Interface.....	27
Figure 18. Data Input Frame	28
Figure 19. Hypothesis Selection Window	28
Figure 20. Value Input Window	29
Figure 21. Data Display Frame	29
Figure 22. Data Graph.....	30
Figure 23. Large Model Approach.....	31
Figure 24. I2AT Approach.....	31
Figure 25. Pre Sequence Threat of War.....	32
Figure 26. Asset Reliability	33
Figure 27. Threat of War scenario with error propagation.	34
Figure 28. System architecture for web-based data validation.	35
Figure 29. Web pages returned by Dr. Knowledge.	36
Figure 30. Entering a variable value in Dr. Knowledge.	37

Figure 31. Updated probability of claim.....	38
Figure 32. Nodes of WMD Nonproliferation Bayes net model.....	40
Figure 33. States and Probabilities of WMD Nonproliferation Bayes net model	41
Figure 34. Threat of War Bayes net model.....	44
Figure 35. Threat of War Bayes net model with states and probabilities	45

List of Tables

Table 1. Utility matrix for decisions regarding radar hypothesis	14
Table 2. Performance of Adaptive Learning vs. Bayesian Updating	25
Table 3. Performance of Model with Bias & Effective Sample Size Error Models	26

1 EXECUTIVE SUMMARY

BAE Advanced Information Technologies (AIT) has developed an Information Interpretation and Analysis Toolkit (I2AT), providing a set of analytic capabilities that fit within and augment a knowledge management system. The I2AT tool will address the challenge of effectively analyzing vast amounts of information with limited analyst manpower by focusing analyst attention on available pieces of information that produce significant changes in the assessment of the situation and identifying additional information that has the potential to do so.

The central capabilities of the I2AT include:

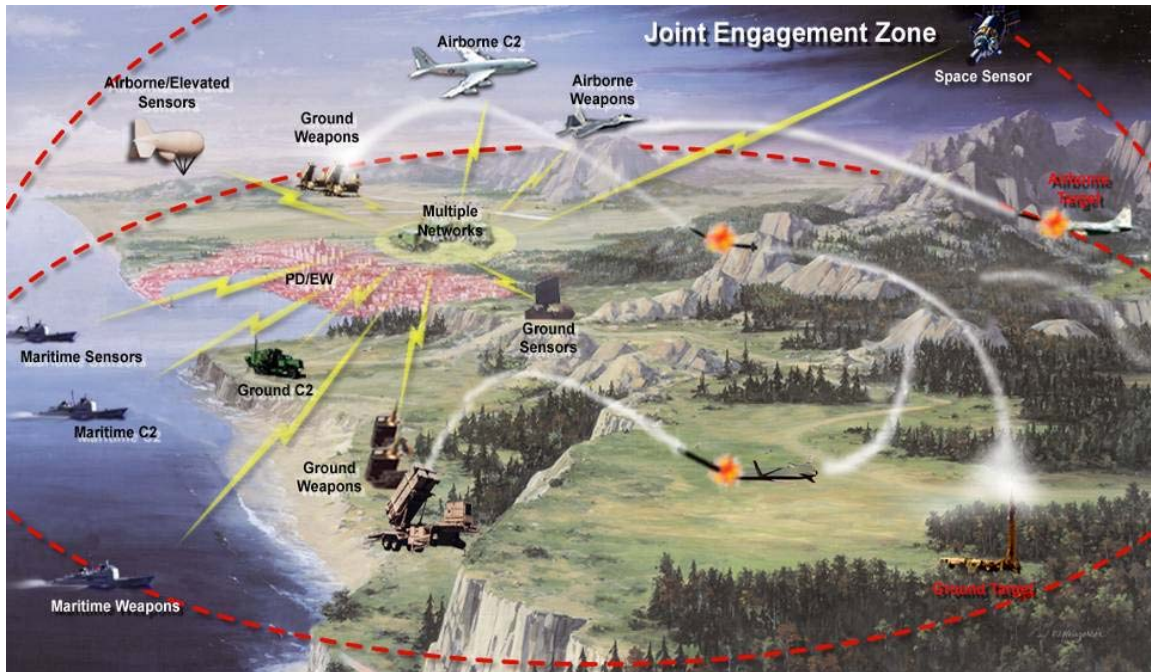
- *Interpretation*: Assisting in the interpretation of sets of information using knowledge-based models of both normal and threat activities.
- *False information detection*: Identifying pieces of information that are inconsistent with the overall information set, whether entered through errors in analysis or through deliberate deception.
- *Data needs generation*: Identifying additional pieces of information that are critical for resolving ambiguities in the interpretation of current information sets. This can be used to ensure effective application of ISR collection and analysis assets, whether automated sensors or human analyst resources.

Our approach uses Bayesian Networks, a probabilistic modeling framework, augmented with a suite of algorithms for analyzing hypotheses, data, and value of additional information. The techniques we have developed are applicable to a wide range of probabilistic models and to a wide variety of domain models.

Automated generation of potential interpretations will help reduce the time needed to assimilate and act on new information. At the same time, flagging new data that is inconsistent with existing information will identify potential errors in the knowledge acquisition process or adversary attempts at deception, enabling preemptive correction before erroneous interpretations precipitate actions. Within a knowledge management framework, data that is tagged as potentially deceptive can be tracked to determine what analytic results need to be called into question. Finally, I2AT will be able to determine which additional information would have the greatest potential to explain observed inconsistencies.

2 INTRODUCTION

2.1 THE INFORMATION INTERPRETATION PROBLEM



Integrated knowledge intensive systems combine data from diverse sources to provide a comprehensive picture of some portion of the environment. In interpreting the output of such systems, a key question is how they handle incorrect or unreliable data. Will the system simply fail to generate reliable interpretations when given faulty data or will it be able to detect faults in particular data items and ignore them when generating hypotheses? This is a crucial question for the application of such systems to real world problems, since we are rarely guaranteed absolutely reliable data in real life situations. The answer to this question influences the degree of confidence we should assign to interpretations by the system and also influences future data acquisition (e.g., if certain data are suspect, there may be further data that could be gathered to resolve ambiguities).

2.2 TECHNICAL CHALLENGES

One challenge is detecting that there is a problem with data in the first place. If two information sources are reporting on the same event and they disagree, then we know that one of them is incorrect. However, in information fusion, the information sources are often reporting on different features of the environment that are indirectly related to the hypothesis of interest. Consider, for example, battle damage assessment regarding the success of an attack on an enemy radar facility. One source of information might be reports by humans visually inspecting the damage; another source of information might be ELINT reporting on whether a radar signal has

been detected coming from the damaged target. If a human source reports extensive damage while an ELINT sensor reports detection of a radar signal, how should we interpret the evidence? Was the human exaggerating the extent of damage? Was the sensor reporting a false positive? Or was the radar still functional despite the extensive damage to the radar facility? To determine which interpretation is correct, we need some way of determining whether there is reason to question the data and if there is, we need to pinpoint which data source is faulty.

The first challenge gives rise to a second: learning an error model for a particular information source. Good error models can help to determine which information source is likely to be in error when there is a conflict among data sources. However, we often lack accurate error models. This is especially true with human sources but can also be true for mechanical sensors, whose operating properties can change over time or may differ in an operational setting from what was measured in a test environment. We need, first, to be able to represent the fact that there is uncertainty regarding the error model for an information source; and second, we need to be able to learn the error model for a source dynamically and automatically, based on how it has performed in the past – and we must often do so *without knowing ground truth*.

A third challenge is how to resolve ambiguities in the data when we are unable to definitively pinpoint which data source is wrong. For example, when the human battle damage report and the ELINT sensor seem to conflict as they do in the example above but we're unable to determine which one, if either, is misleading, we may seek to resolve the impasse by obtaining additional information. Which information would be most useful, however, in resolving the ambiguity? Should we request another fly-over of the damaged target, which may be costly and also risky if the radar is still functioning? Should we apply another sensor to the target, say, a stationary rotor sensor that detects the motion of a radar antenna? These are questions about the value of particular information sources in the situation at hand. A value of information computation is needed that takes into account the possible unreliability of information sources.

Finally, we must not forget the ultimate goal of information interpretation is to provide a unified picture of the environment. The output of an information interpretation system is a *hypothesis* about some aspect of interest of the operational environment. Proper interpretation of the information involves not just the production of a hypothesis but also a determination of how much confidence we should place in the hypothesis. Probabilistic interpretation systems, such as Bayes nets, attach a number to an assertion that acts in some sense as a measure of confidence in a hypothesis (e.g. "The probability that the radar has been destroyed is 0.78"). However, a single number by itself does not tell the whole story. Is the number based on the accumulation of large amounts of data or only a small amount of data, is it based on reliable information sources or unreliable ones, might the number change significantly if further data comes in? These questions relate to a notion of hypothesis confidence that cannot be captured by a single number but their answers are vital in assessing the hypothesis.

2.3 THE I2AT SOLUTION

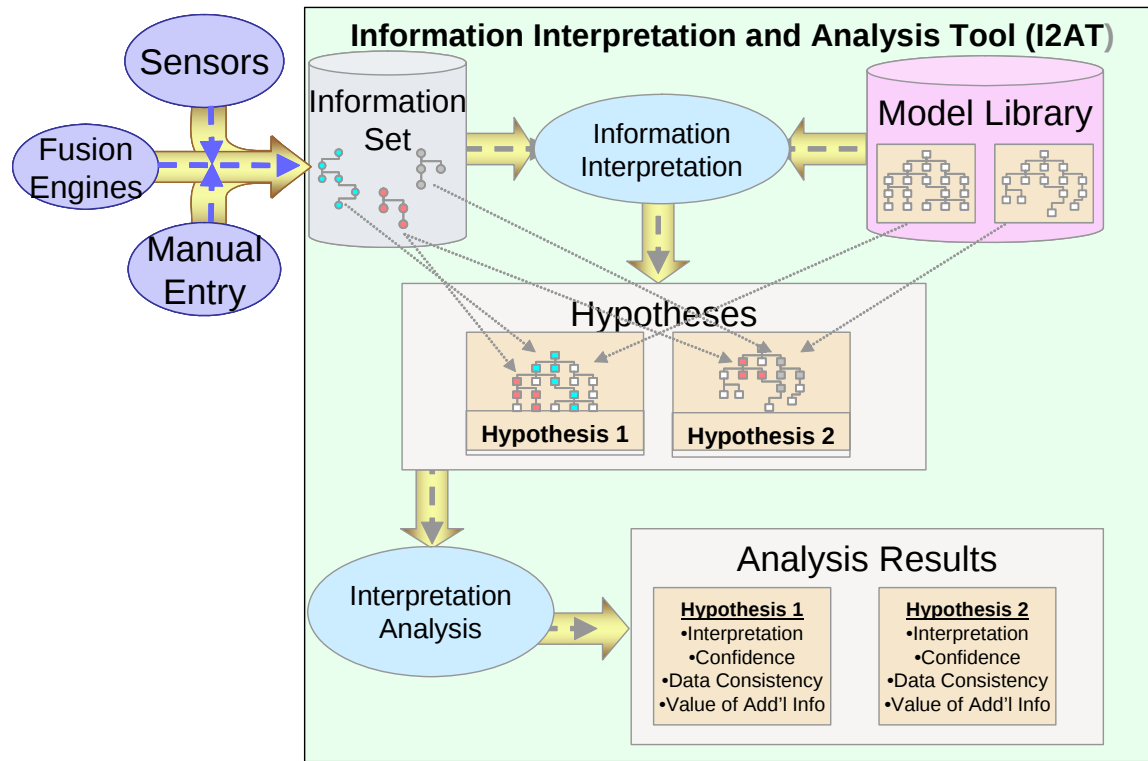


Figure 1. I2AT architecture.

The I2AT solution to the data validation problem is shown in Figure 1. We have implemented a complete prototype system for information interpretation that demonstrates the utility of our data validation tools. Information from various sources may be entered into the system. Data may come from sensors, manual entry, or backend fusion engines. There are two stages to the processing of that information. In *information interpretation*, we use a library of models to generate *hypotheses* from the data. After hypotheses are generated, we then perform *interpretation analysis*, which goes beyond the generation of hypotheses by computing a collection of statistics that help the user understand more fully the status of both hypotheses and data and indicates whether or not there is a need to collect more data. The statistics computed include:

- *Data conflict*: Are the data in conflict? Conflicts in the data indicate that there may be problems with certain data items.
- *Hypothesis confidence*: For a probabilistic hypothesis, this measures the degree to which the probability will change, on average, as new evidence comes in. A probabilistic hypothesis has high confidence if new evidence will not shift the probability to any significant degree.

- *Data confidence*: For each data item, this measures the probability that that data item is correct. If a data item has low confidence, I2AT will accord it little weight in forming a hypothesis.
- *Weight of evidence*: This is a measure of the degree to which a particular piece of evidence supports or opposes the hypothesis. This gives an indication of which pieces of evidence are most important in judging the truth of the hypothesis and which have little influence on it.
- *Value of information*: If existing data are in conflict and hypothesis confidence low, new information should be gathered to resolve the uncertainties. Value of information measures which new piece of information would best resolve remaining uncertainties about the hypothesis.
- *Reliability of data sources*: Our confidence in the data produced by some data source on a particular occasion should influence our assessment of its reliability. Repeated generation of bad data should cause us to lower our assessment of a source's reliability. I2AT updates the reliability of data sources based on what that data source reports and whether it agrees with what other data sources are reporting.

Key to I2AT's ability to compute the above statistics is our approach to modeling. We make use of probabilistic models – in particular, Bayesian Networks ([3], [5]). Our probabilistic models differ from most models in that we make use of explicit variables in the model for the reliability of evidence sources. We are in effect introducing an *error model* for information sources into the Bayes net that allows us to reason about the correctness of particular reports and what new evidence would best resolve ambiguities.

Section 3 describes our modeling technique in more detail. Section 4 explains the full range of statistics calculated by I2AT and how they are computed. Section 5 surveys previous work on data validation and compares our approach with this work. Section 6 reports on experiments in learning error models using two different learning algorithms. Section 7 describes the user interface to I2AT. Section 8 explains how I2AT can handle inference across multiple scenarios and involving multiple Bayes nets. Sections 9 and 10 describe how I2AT can be applied to domains that are quite different from the BDA domain on which we have focused for most of the program. We conclude in Section 11 with an exploration of ways in which the techniques developed in I2AT can be extended.

3 MODELING IN I2AT

To describe our approach to modeling, it is useful to have a concrete scenario in mind. We will describe a targeting scenario relevant to effects-based operations and show how we model it in I2AT. Crucial to the model will be sub-models for information sources.

In our example scenario, our side has carried out an air strike on an enemy radar facility. A battle damage assessment is performed to determine whether the radar was really destroyed. There are multiple sources of information. We have BDA reports from the two pilots who

carried out the air strike. The two pilots give their assessment of the amount of damage to the target. In the case of radar operation, we have two sensors that provide different indicators. The ELNOT sensor detects emissions from the radar (True or False). The Rotor sensor detects whether the radar is rotating (True or False). Both sensors give a sequence of reports that are temporally ordered.

We use a Bayesian network to model this scenario. A Bayesian network is a directed acyclic graph consisting of nodes which represent random variables ([3], [5]). They allow for a compact representation of complex probabilistic dependencies. In a discrete BN, each node of the graph must also have a conditional probability table (CPT) associated to it. A CPT specifies the exact local probabilistic dependencies (in particular, how the states of the variable probabilistically depend on each possible configuration of the parent variables). All of our models are discrete Bayesian networks.

Bayesian networks are useful in data validation and interpretation. They are well-suited to capturing expert knowledge due to their modularity and understandability. In addition, learning from historic data, automating interpretation, deducing the most likely true states and most likely errors, and determining the reliability of sources are tasks that BNs perform well. They can also focus the analyst's attention on the most important pieces of information, highlight inconsistent, potentially incorrect data, and point out information which may resolve ambiguities ([3]).

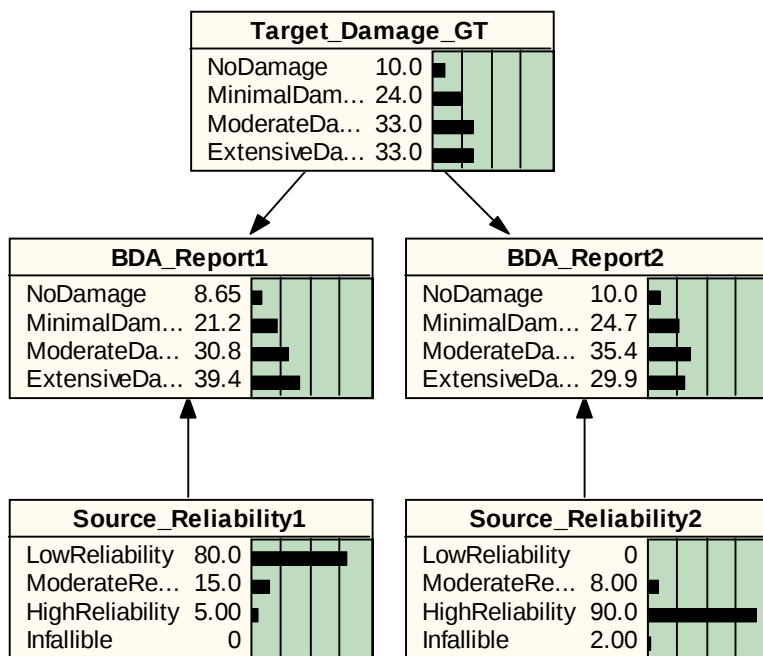


Figure 2. Bayes net fragment for pilots' reports.

We use Netica™, a popular Bayes net toolkit, to create our models. Figure 2 shows a fragment of a Netica Bayes net relevant to our BDA scenario. It models the relationships between what the pilots report, ground truth about target damage, and each pilot's reliability. The variables in

the net are shown as rectangles with the variable name at the top followed by the states the variable can assume together with their prior probability (the probability is expressed as a percentage, so the value 33 is a probability of 0.33). The directionality of the arrows shows that what each pilot reports is influenced by the actual damage to the target as well as by the reliability of that pilot. The report of the first pilot (BDA_Report1) is apt to be more inaccurate than the report of the second pilot (BDA_Report2) due to the higher probability that the first pilot is unreliable in his reporting. Moreover, the first pilot's reports are biased in the direction of exaggerating damage, whereas the second pilot's reports, although not guaranteed to be accurate, tend to report damage without much bias.

Note that the reliability of each pilot is characterized as a probability distribution over a range of reliability values. This allows uncertainty about the reliability of a pilot to be expressed and also permits modifying beliefs about pilot reliability.

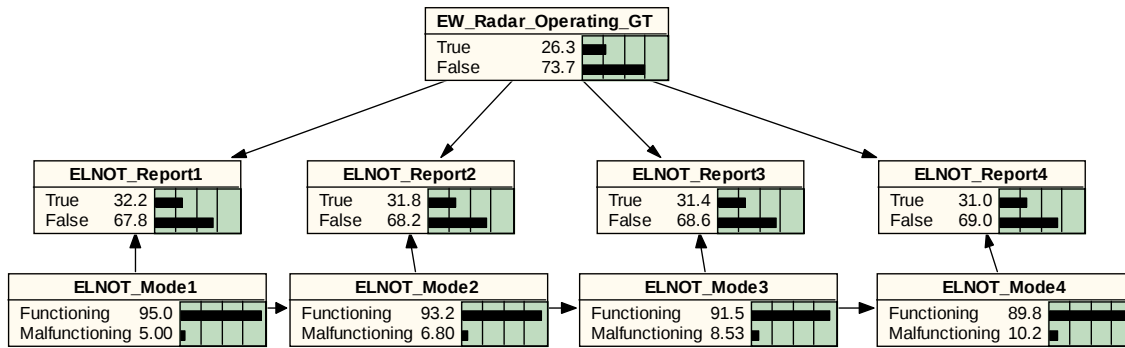


Figure 3. Bayes net fragment for ELNOT sensor.

Figure 3 shows another fragment of our Bayes net model, a representation of an ELNOT sensor. This sensor issues reports about whether it has detected a radar signal. The various **ELNOT_Report_i** variables are reports by the same sensor at four different times (reports variables with a higher number are at a later time). What the sensor reports at any time depends upon whether the radar is operating (putting out a signal) and also on the mode of the sensor, which may be “Functioning” or “Malfunctioning.” The arrows between the mode variables indicate that the state of the sensor at one time depends upon its state at a previous time. If the sensor is malfunctioning at one time, it is likely to still be malfunctioning at the next time step. When the sensor is malfunctioning, it will likely not report any signal but may report a signal present with some small probability; in either case, what it reports when it is malfunctioning is completely independent of the whether the radar is operating.

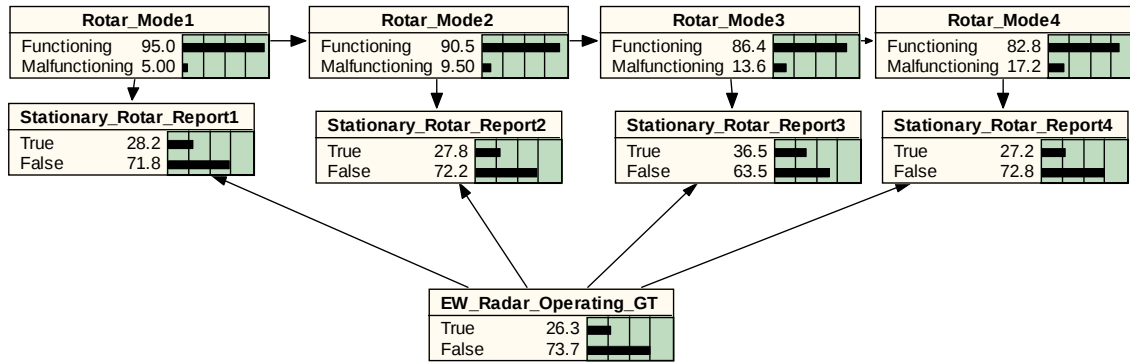


Figure 4. Bayes net fragment for stationary rotor sensor.

A parallel model is given for the Stationary Rotor Sensor, which detects rotary motion of a radar antenna (Figure 4).

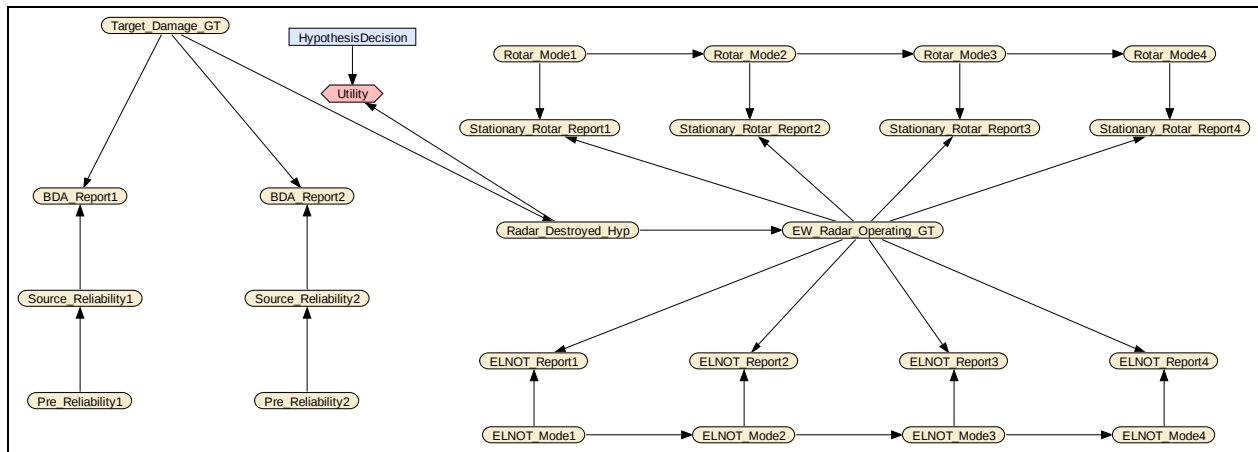


Figure 5. Entire Bayes net for BDA model.

Putting these fragments together, we get the network depicted in Figure 5.. The same network is displayed in Figure 6 below and includes all the node states and corresponding probabilities.

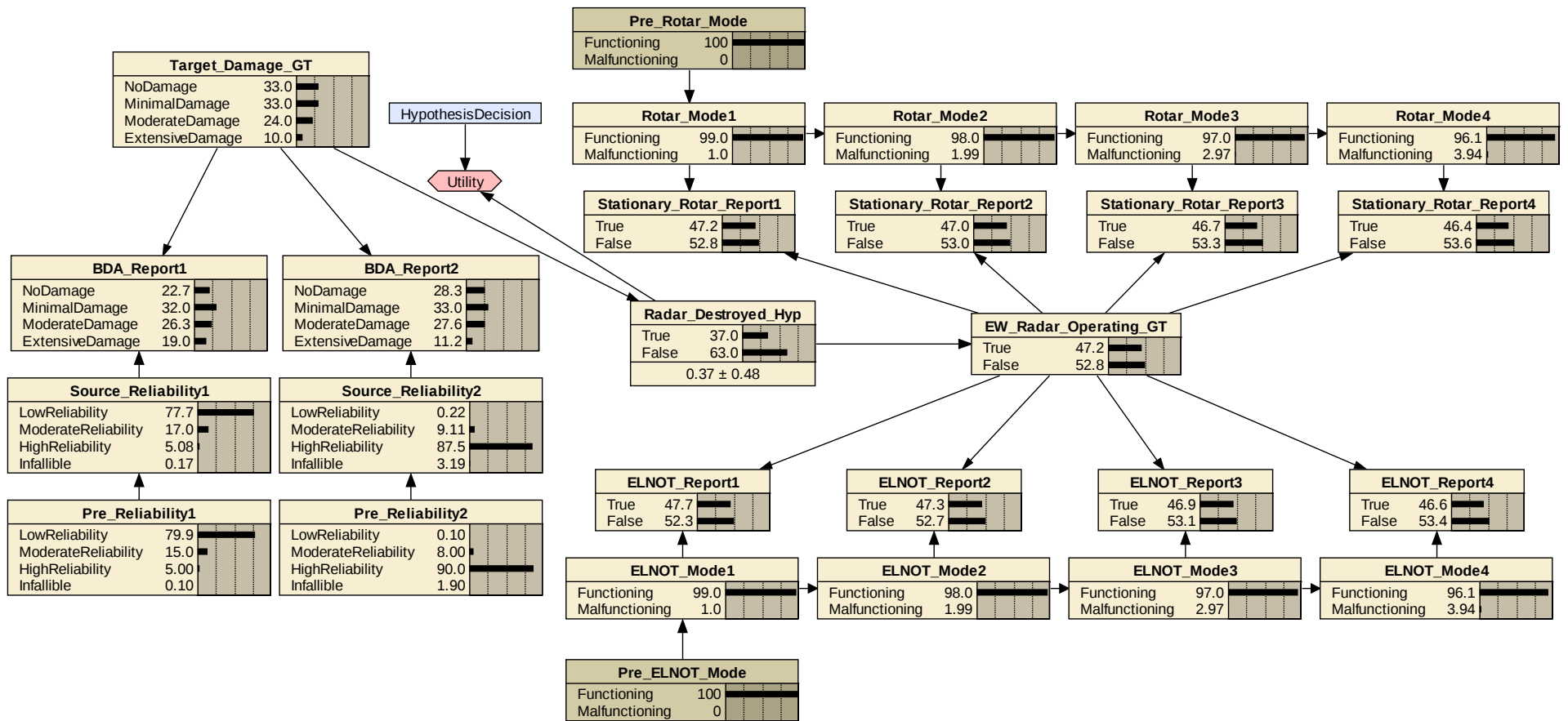


Figure 6. Full States and Probabilities for Bayes net BDA model

4 I2AT APPROACH

This section discusses in detail our techniques for analyzing data and hypotheses.

4.1 DATA CONFLICT

The first step in analyzing the data is to determine whether there is any conflict in the data. Conflict in the data indicates that one or more pieces of evidence may be faulty. A technique for measuring the amount of conflict in data has been given by Jensen in [3].

Jensen's measure of data conflict is as follows. Let e_1, e_2, e_n be data items – i.e. assignments of values to evidence variables. Jensen defines a measure of data conflict by:

$$\text{conflict}(\{e_1, e_2, \dots, e_n\}) = \log((\Pr(e_1)\Pr(e_2)\dots\Pr(e_n))/\Pr(e_1, e_2, \dots, e_n)).$$

Why does this measure make sense? $\Pr(e_1)\Pr(e_2)\dots\Pr(e_n)$ is what the joint probability of $e_1, e_2, \dots, e_n$ *would be* were all the data items probabilistically independent of one another; $\Pr(e_1, e_2, \dots, e_n)$ is their actual joint probability. Jensen's conflict measure in effect compares the joint probability of the data expected under an independence model with their joint probability under the actual model. If the former is significantly greater, that is very suspicious. Why? Because if some of the evidence supports a given hypothesis, it's likely the rest of the evidence will too. That is, given some evidence that favors one hypothesis, it is likely that other pieces of evidence will too. Thus the new evidence one gets is very unlikely to be *independent* of the previous evidence. It's more likely to be made more probable by the previous evidence. So in the normal case, evidence will be *harmonious* or *mutually supportive*. If there is a strong degree of conflict among the evidence, then either a highly unlikely combination of evidence events has occurred or else *one or more pieces of evidence are wrong or misleading*. If the probability of that combination of evidence is low enough, we are led to suspect problems with the evidence.

To illustrate the computation of data conflict, suppose that pilot1 reports extensive damage to the target while pilot2 reports minimal damage. Intuitively, there is a conflict in these reports. To show this, we note that we have from Figure 2 that $\Pr(\text{BDA_Report1} = \text{ExtensiveDamage}) = 0.274$ and $\Pr(\text{BDA_Report2} = \text{MinimalDamage}) = 0.247$. If these two pieces of data were probabilistically independent of one another, then the probability of getting both reports would simply be the product of their prior probabilities – i.e. 0.274×0.247 or approximately 0.0677. Using the Bayes net, we can compute that the probability of getting both reports is actually 0.0433. $\log(0.0677/0.0433) = \log(1.564) = 0.447$. This number is positive, so there is conflict in the reports, as we expected.

We note that the computation of data inconsistency can be done even when the model does not contain any explicit error variables. This allows data inconsistency to be computed even when error models are not available.

This measure of conflict only serves to indicate when there is a problem somewhere in the data. It doesn't tell us which data item is problematic. To pinpoint a faulty data item, we can use a modified version of the conflict measure, where now conflict is measured between two subsets of data items:

$$\text{conflict}(D_1, D_2) = \log((\Pr(D_1)\Pr(D_2))/\Pr(D_1 \cup D_2)).$$

By combining this measure with the original conflict measure over sets of data items, we can obtain a better idea of where problematic data are located. If, for example, $\text{conflict}(D_1)$ is negative and $\text{conflict}(D_2)$ is not, then a high value for $\text{conflict}(D_1, D_2)$ strongly indicates that the problem lies in D_2 . If D_2 has only one element, then we have pinpointed the faulty data.

This criterion was tested on several cases involving the BDA model. When a single report has a high degree of conflict with the remaining data items, it does indeed turn out that the degree of confidence in that report, as calculated using the error model, is low.

4.2 DATA CONFIDENCE

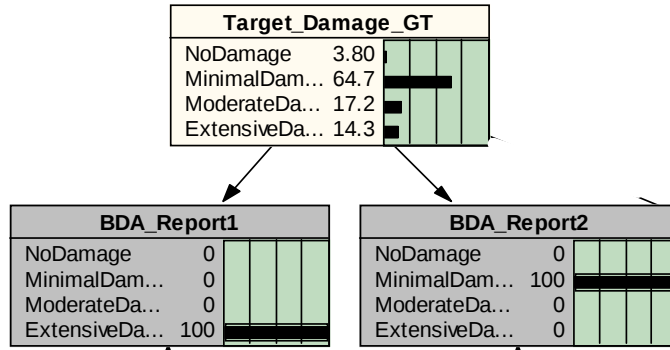


Figure 7. BDA model with BDA reports entered as evidence.

To compute data confidence, we compare what is reported by an information source with the probability of the corresponding ground truth variable given all the available evidence. So for example, when the only evidence we have is that pilot 1 has reported extensive damage and pilot 2 has reported minimal damage, the Bayes net is updated as shown in Figure 7. The Bayes net computes the posterior probability of the states of *Target_Damage_GT* given this evidence as shown above. We see that the probability of *MinimalDamage* is 0.647 while the probability of *ExtensiveDamage* is only 0.143. Our confidence that pilot 2 is right is therefore 0.647 and our confidence that pilot 1 is right is 0.143. We therefore have a much lower confidence in the report of pilot 1 than in the report of pilot 2.

4.3 HYPOTHESIS CONFIDENCE

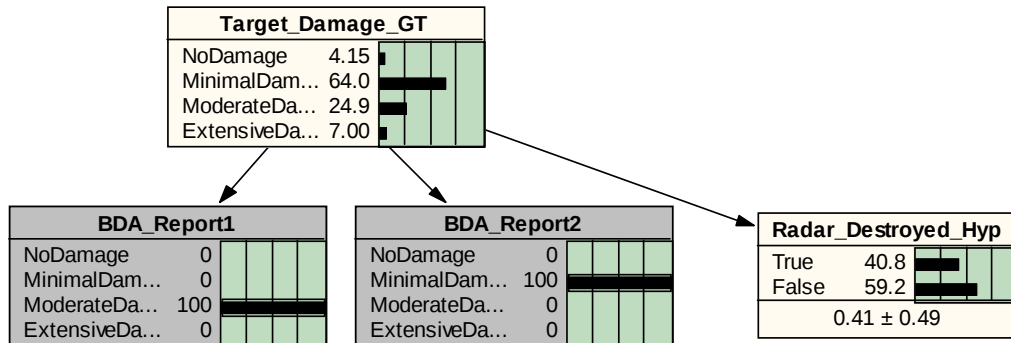


Figure 8. Initial evidence for "Radar Destroyed" hypothesis.

Hypothesis confidence is a measure of how much a probabilistic hypothesis may be expected to change in the light of new evidence. In the example shown in, two pieces of evidence have been entered into the Bayes net: a report of moderate damage from pilot 1 and a report of minimal damage from pilot 2. The hypothesis of interest in this case is the assertion that the radar has been destroyed – the value of the variable `Radar_Destroyed_Hyp` = True. Given the available evidence, the probability that the hypothesis is true is 0.408.

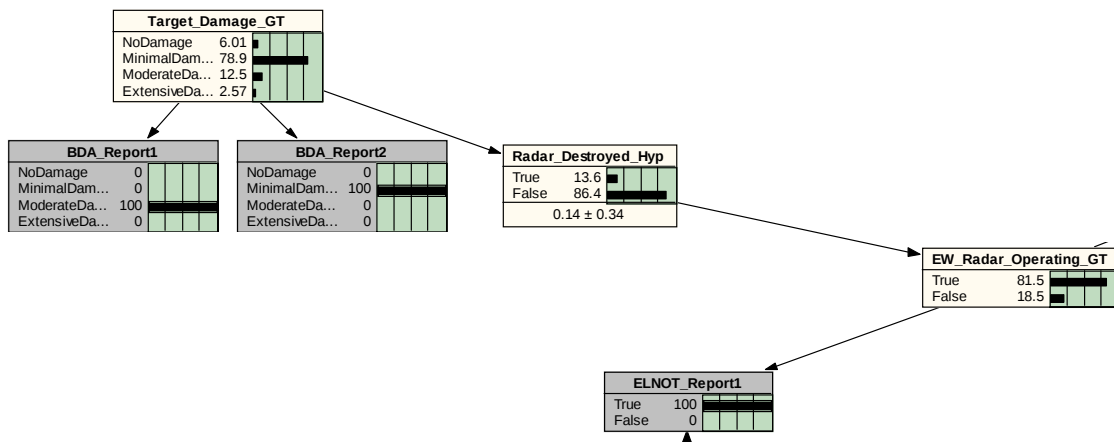


Figure 9. Probability of hypothesis changes with new evidence.

How much confidence should we have in the estimate of 0.408 for the probability that the radar has been destroyed? To answer that question, we look at other potential sources of evidence and see to what degree that probability would change if additional evidence were acquired. For example, we might have a report from the ELNOT sensor. If the sensor reported detecting a

signal (ELNOT_Report1 = True), then the probability that the radar was destroyed would change dramatically, from 0.408 to 0.136. Clearly, then, the current probability of 0.408 is not very robust and we should not place a high degree of confidence in it.

To compute hypothesis confidence, we divide the evidence variables up into two sets. **A** is the set of evidence variables to which assignments have been made (specific values have been entered for them). **U** is the set of evidence variables to which no assignments have been made. This set represents variables for which evidence has not yet been acquired but might be. Let *e* be the assignment of values that has been made to the variables in **A**. *e* is the evidence acquired so far. Let *f* be an arbitrary assignment of values to the variables in **U**. *f* represents potential evidence (or evidence that might be acquired in the *future*). We use assign(**U**) for the set of all assignments of values to the variables in **U**. Then we define the *volatility* of the current probability of hypothesis *h* given evidence *e* by:

$$\sum_{f \in \text{assign}(\mathbf{U})} P(f | e) | P(h | e) - P(h | f, e) |$$

This computes the average difference between the probability of *H* given the current evidence and the probability of *H* given both current and potential future evidence, where the difference is weighted by the probability given the current evidence that we would obtain that potential evidence.

Our *confidence* in a probabilistic hypothesis varies inversely with the volatility of the hypothesis. We may therefore define *hypothesis confidence* by:

$$1 - \sum_{f \in \text{assign}(\mathbf{U})} P(f | e) | P(h | e) - P(h | f, e) |$$

4.4 VALUE OF INFORMATION

If we have low confidence in a hypothesis or there is ambiguity regarding the data, we may wish to obtain further evidence to resolve the uncertainties. We implemented two metrics for determining value of information. One is a purely information-theoretic metric that measures the expected gain in information from querying an unassigned evidence variable; the other quantifies the benefit to decision making of additional information.

The information-theoretic criterion for value of information says to choose as a query an evidence variable for which the mutual information between that evidence variable and the hypothesis variable is maximal. Where *H* is an evidence variable and *E* is an evidence variable, the mutual information $I(H ; E)$ between *H* and *E* is given by:

$$\sum_{h \in H} \sum_{e \in E} \log \left(\frac{\Pr(h, e)}{\Pr(h) \Pr(e)} \right)$$

This is 0 if *H* and *E* are completely independent and positive if *H* and *E* are correlated.

The second measure of value of information uses the criterion of expected utility to choose an evidence variable for querying. Use of this criterion requires that we be able to specify a set of

decisions related to the hypothesis variable of interest and a utility or payoff for making a given decision for each value of the hypothesis variable.

Table 1. Utility matrix for decisions regarding radar hypothesis.

	Accept Radar Destroyed	Reject Radar Destroyed
Radar Destroyed	10	-10
Radar not Destroyed	-100	0

In the case of our BDA scenario, the hypothesis of interest is whether or not the enemy radar has been destroyed. We can take the decisions to be decisions as to whether or not to accept the hypothesis that the radar has been destroyed. A matrix showing the utility of each decision for each possible ground truth state is shown in

Table 1. Correct decisions have a higher utility than incorrect decisions. The utility recorded here is overall utility, not just the increment to utility from making a given decision. So the utility of accepting that the radar has been destroyed when it has been destroyed is higher than the utility of rejecting that the radar has been destroyed when it hasn't been destroyed, because the former outcome is a better one overall than is the second one. (We strongly prefer that the radar be destroyed.) The worst possible outcome is that we mistakenly think the radar has been destroyed when it has not been destroyed.

The numbers in the table above are for illustrative purposes only. In a fielded system, a careful cost/benefit analysis would have to be done in consultation with subject matter experts to determine appropriate values for the table.

Given the utility matrix, we can then define the *expected utility criterion* for value of information as follows. As before, let H be the hypothesis variable and E an unassigned evidence variable. And let D be the set of decisions regarding the value of variable H . For each value h of H and decision d in D , let $U(h,d)$ be the utility of making d when h is the value of H . Then the expected utility of querying variable E is given by:

$$\sum_{e \in E} \Pr(e) \max_{d \in D} \left\{ \sum_{h \in H} \Pr(h | e) U(h, d) \right\}$$

The expected utility criterion says to choose for querying the variable E than maximizes the above function.

It will often happen that the information-theoretic and expected utility criteria for information value agree. However, in experiments, we have found cases in which they disagree on which variable to query next.

We note that the both value of information criteria are *myopic*. That is, they only look at what the best variable to query would be if that were the only additional variable one could query. They do not consider the utility of querying sequences of variables.

4.5 WEIGHT OF EVIDENCE

Another data statistic useful in interpreting the evidence is which data items support the hypothesis (make it more probable) and which ones oppose it, and the degree to which they do so. Data all of which strongly points in one direction should give us confidence in the hypothesis pointed to, but when there is equal evidence of equal weight pointing in different directions, we should be unsure what the correct hypothesis is. Moreover, if a data item has negligible weight either for or against the hypothesis, then we may conclude that the source of that data is not informative with respect to hypotheses of that nature and ignore it or not consult it in the future.

Let the evidence E be the set $\{e_1, e_2, \dots, e_n\}$ (each e_i is an assignment of a value to an evidence variable). Where h is the hypothesis and $\text{not-}h$ is the negation of the hypothesis (the assertion that the hypothesis variable does *not* have the value specified by h), we define the *weight of evidence of e_i for h* by:

$$\text{weight}(e, h, E) = \log \left(\frac{\Pr(e \mid h, E - \{e\})}{\Pr(e \mid \text{not-}h, E - \{e\})} \right)$$

This expression compares how likely the evidence e would be given the hypothesis and the rest of the evidence with its likelihood given the negation of the hypothesis and the rest of the evidence. So weight of evidence for a piece of evidence is computed relative to what the rest of the evidence is. When $\text{weight}(e, h, E) > 0$, e supports h in the context of E ; when $\text{weight}(e, h, E) < 0$, e opposes h in the context of E .

4.6 ERROR MODELING

This section describes modeling techniques for representing information source error that we developed in the course of the I2AT project. These modeling techniques were tested on Netica models but were not fully exploited in our main BDA scenario model.

To capture uncertainty about the error model for a given source, we make sources of error explicit variables in the Bayes net model and place a probability distribution over their values. An obvious example of this is to make the reliability of a source a variable with, for example, values “LowReliability,” “ModerateReliability,” and so on, and to assign prior probabilities to these values. This allows our degree of belief in the reliability of a given source to be modified by what other sources are reporting. For example, we might initially consider it probable that a

particular source is highly reliable but change our opinion if the source is contradicted by multiple other sources that we believe to be highly reliable.

Reliability, however, is not the only variable relevant to modeling source error. Bias is also important. Reliability can be defined as the probability that the source will give a correct answer; bias, on the other hand, is a measure of the direction in which a source errs when the source does make an error. Knowledge of the bias of a source can give us valuable information when we attempt to infer something from what the source reports. Suppose, for example, a source tells us that damage to a certain target was minimal. If we regarded this source as of low reliability, we wouldn't get much information from this report. The damage might actually be minimal but given the unreliability of the source, it might very well be more extensive or it might be none at all. Suppose, however, that we also know that this source is biased in the direction of exaggerating damage. We still think the source is not that reliable, but given our knowledge of the source's bias, we can make a more precise inference that the damage to the target probably is minimal or is none at all.

In addition to modeling bias, we investigated another kind of problem with data, namely, missing data. Sometimes the fact that data is missing is completely uninformative – if the ELNOT sensor has not been activated and we know that, the fact that we have no report from it is no reason to revise our probabilities about whether the enemy radar is operating. On the other hand, if we fail to receive a battle damage assessment from the pilot who dropped the bomb and we were expecting to receive one, that may indicate that the pilot has experienced some kind of difficulty and we should decrease our confidence in his having successfully completed his mission. We discuss these new error modeling features in more detail below.

Defining bias.

Bias can be defined in a number of ways. We will use the technical definition of “bias” in statistics. Let X be a real-valued variable (this includes variables with integer values) and let Y represent an estimated value for X based on, e.g., a sensor measurement. The *bias* of Y is defined to be $E(Y - X)$, the expected difference between the value of Y and the value of X . From the definition of expectation, we have

$$\begin{aligned}
 E(Y - X) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} P(X = x, Y = y)(y - x) dx dy \\
 &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} P(X = x, Y = y) y dx dy - \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} P(X = x, Y = y) x dx dy \\
 &= \int_{-\infty}^{\infty} P(Y = y) y dy - \int_{-\infty}^{\infty} P(X = x) x dx \\
 &= E(Y) - E(X),
 \end{aligned}$$

where $P(\cdot)$ is a probability measure over the space $X \times Y$.

So the bias of a measurement variable Y with respect to a ground truth variable X is the difference between the mean of Y and the mean of X .

This definition of bias can be applied to binary variables (such as RadarOperating). To do so, we represent the truth-value true as the numerical value 1 and the truth-value false as the value 0. If, then, the radar is operating, we have RadarOperating = 1; otherwise, RadarOperating = 0. A report variable ReportRadarOperating will have the value 1 if it is reported that the radar is operating and 0 otherwise. In this case, we have

$$\text{bias}(\text{ReportRadarOperating}) = P(\text{ReportRadarOperating} = 1) - P(\text{RadarOperating} = 1).$$

That is, the bias in the report is the difference in the prior probability of reporting that the radar is operating and the prior probability that the radar is operating.

Error models with both reliability and bias. Suppose we have some variable such as TargetDamage that can be associated with a numerical scale, say, 0 for no damage and 10 for complete destruction. We can always rescale so that the values fall in the range [0,1] (e.g. the value of 1 corresponds to the value of 10 on the old scale, 0.3 to 3, and so on). Then we may ask: given that the true value of TargetDamage is x , what is the probability that the source will report a value close to x ? More generally, can we specify a probability distribution for report values conditional on each value of TargetDamage?

One way to do so is to express the conditional distribution for the report variable as a function of the TargetDamage value. Given that TargetDamage = x , the value of TargetDamageReport could be any value between 0 and 1, with values close to x being more probable the more reliable the source of TargetDamageReport is. One way to express such a conditional distribution is through the *Beta distribution*, which is a parameterized probability distribution over the interval [0,1]. Where a, b are the parameters of the Beta distribution, we define $\text{beta}(y; a, b)$ for y in the interval [0,1] as

$$\frac{1}{B(a,b)} y^{a-1} (1-y)^{b-1}$$

$B(a, b)$ is the Beta function and ensures that integration of $\text{beta}(y; a, b)$ from 0 to 1 will yield 1 (so that $\text{beta}(y; a, b)$ is a genuine probability density function).

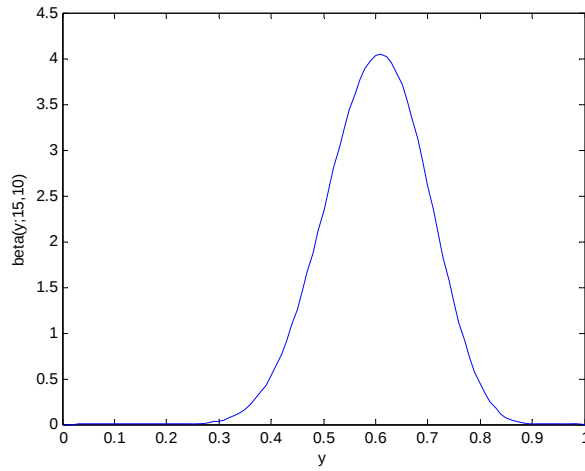


Figure 10. Beta pdf for $a = 15$, $b = 10$.

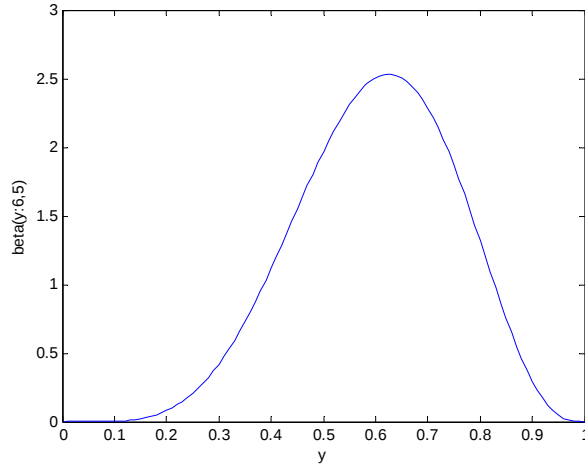


Figure 11. Beta pdf for $a = 6$, $b = 4$.

Figure 10 and Figure 11 show two Beta distributions with different values of a and b . The two distributions have the same mean: 0.6. The mean of a Beta distribution is $a/(a + b)$ and the variance monotonically increases as $a + b$ decreases. Visual inspection of the two distributions verifies that the second distribution is much more spread out than the first. If the two distributions represent the probability that a source will report a value y given a ground truth value of x^* , then the source for the second distribution is more “erratic” than the source for the first.

Each of the above Beta distributions represents the probability that the report variable Y will assume a certain value conditional on the ground truth variable having a particular value x^* . A model for generating such distributions above given a value x^* of the ground truth variable X is as follows. Fix the sum $N = a + b$. Pick some number f such that $-1 \leq f \leq 1$ and determine a by:

$$a = \begin{cases} Nx + fNx & \text{if } f < 0 \\ Nx + fN(1-x) & \text{if } f \geq 0 \end{cases}$$

Since the mean of a Beta distribution with parameters a, b is $a/(a+b)$ or a/N , this definition entails that the mean of Y is given by:

$$E(Y | X = x) = \begin{cases} x + fx & \text{if } f < 0 \\ x + f(1-x) & \text{if } f \geq 0 \end{cases}$$

Since we have:

$$E(Y) = \int_0^1 P(X = x)E(Y | X = x)dx,$$

we see that if $f = 0$, $E(Y) = E(X)$ and Y will be an unbiased estimator of X in that case. If f is negative $E(Y)$ will be less than $E(X)$ (unless $P(X = 0) = 1$), so Y will have a negative bias; similarly, if f is positive, $E(Y)$ will exceed $E(X)$ (unless $P(X = 1) = 1$) and Y will have a positive bias.

We call f the *bias factor*. If we set $N = 25$ and $f = 0.2$, we get the Beta distribution shown in Figure 10 for $X = 0.5$. If we let $N = 10$ and keep f at the value 0.2, we get the Beta distribution shown in Figure 11 for $X = 0.5$.

Note that the bias factor is not the bias. If X is uniformly distributed over the interval $[0,1]$, we can show that the bias of Y is $f/2$.

Δ -reliability. For any realistic error model, the probability that a report will give *exactly* the correct value for a *continuous* variable is zero. Therefore, on the definition of reliability as the probability of giving a correct value, reports on continuous variables have zero reliability. Obviously, however, some measurements of continuous variables are more reliable than others in the sense of having a greater probability of coming close to the true value. We therefore introduce the concept of *Δ -reliability*, which is the probability that a measurement will come within Δ of the true value. If $P(\cdot)$ is a probability measure over the space $X \times Y$ (where X is a continuous ground truth variable and Y is a measurement of X , with both X and Y in the range $[0,1]$), we define the Δ -reliability of Y with respect to X by:

$$reliability(Y, X, \Delta) = \int_0^{\Delta} \int_0^{\Delta} P(x, y) dy dx + \int_{\Delta}^{1-\Delta} \int_{x-\Delta}^{x+\Delta} P(x, y) dy dx + \int_{1-\Delta}^1 \int_{1-\Delta}^1 P(x, y) dy dx$$

Independent error model variables. A drawback to using reliability and bias as variables in modeling source error is that they are not independent variables. That is, it is impossible for a source to have some bias and at the same time be perfectly reliable – bias implies unreliability. Ideally, we would like the variables used in modeling errors to be independent of one another, so that probability distributions over each can be independently specified. Going back to the Beta distribution, we can see that the sum $N = a + b$ can be specified independently of the bias factor f but that the two together determine the distribution for the report variable Y and hence determine the reliability and bias of Y . A Beta distribution is often thought of as giving the probability of

a probability – i.e. given an event of unknown probability p and a sample of events a of which are instances of the event and b of which are not, what is the probability that p is a particular value in $[0,1]$? On this interpretation, $N = a + b$ is the size of the (imaginary) sample. Accordingly, we call variables for the value of N “sample size” or “SS” variables, for short.

Discretizing sample size and bias. Since Netica’s belief propagation algorithms work only for discrete variables with a finite number of values, we need to bin values of sample size and bias. We let the values of sample size be “very small,” “small,” “medium,” “large,” and “very large.” These values correspond to sample sizes of 4, 8, 16, 64, and 128, respectively. Other sample sizes get thrown into a bin containing the nearest number in this sequence. Similarly, the values of the bias variable are “LargeNegative,” “SmallNegative,” “None,” “SmallPositive,” and “LargePositive.”

5 PREVIOUS WORK IN DATA VALIDATION

Data validation work has been performed in database management. Human entry is typically the source of errors. The data values, relations, and data relevance may be unknown, imprecise, or wildly inaccurate. Data validation techniques often involve the forcing of hard constraints (such as constraining age to be a positive number), identifying outliers based on a specified statistical model, creating fuzzy rules, or attaching pedigree information on data [Parsons, 1996]. These approaches mostly target gross errors in collecting or entering data. Such methods are well-suited to relational data in which the scope of data dependencies is limited. However, they may fail to notice improbable combinations of reasonably likely values (and even if it did, it might have difficulty determining which of the values are accurate).

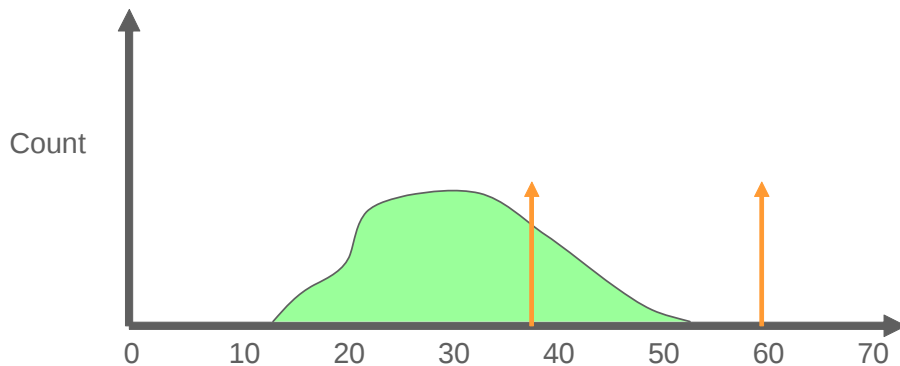


Figure 12. Distribution of ages of terrorists.

In the field of statistics, there are techniques for determining “outliers” or data points that appear anomalous in the light of the rest of the data or in the light of known distributions. For example, assuming we know that the distribution of ages of terrorists is as given in Figure 12, we can say that the assertion that a particular terrorist’s age is 38 would be unsurprising whereas the assertion that his age is 60 would be quite surprising. We might suspect a mistake, then, in a report in which a terrorist’s age is given as 60. This technique can be extended to handle

multiple variables, but it suffers from the limitation that the joint distribution over the variables must be known. Moreover, it is not clear how it would be extended to handle the complex relationships among discrete variables with which we deal in I2AT.

There has been a great deal of work on data validation for sensor fusion (SF). This work is probably closest in spirit to our approach but differs in significant ways. This typically employs fixed, probabilistic noise models which are known a priori. This approach is suited for fusing potentially inaccurate or missing data in a systematic manner. It assumes that we have models for the types of errors made by the sensors, typically in the form of a specification of probability of detection and false alarm probability (probability of a false positive). Differences from our approach include:

- SF is not usually concerned with probability of error on individual sensor readings whereas we might be.
- SF is usually based on fixed, engineered noise models whereas we want to allow both expertise and data to constrain noise models.
- SF often combines information sources with domain-specific methods based on first-principles engineering (e.g., for extracting distance from stereo disparity in two images) whereas we focus on domain-general methods.
- SF focuses largely on combining information from multiple sensors of the same type whereas we want to accommodate heterogeneous information sources.

Data validation work has been performed in law. In particular, there is a long tradition of modeling human testimony probabilistically. The field has spawned a significant amount of literature on reasoning about human testimony. Human testimony can suffer from unintentional biases. However, there does not appear to be a widely accepted method of modeling these biases [6].

6 LEARNING ERROR MODELS

In this section, we present a method of testing our data validation framework. We start by describing the learning algorithms. Afterwards, we explain the testing framework. We close the section by noting how our data validation approach would be applied in practice.

6.1 LEARNING FRAMEWORK

Before we discuss the testing, we need to talk about how we perform learning. We focus on online, incremental learning algorithms which adjust the parameters as data comes in on a case by case basis. Furthermore, our learning algorithms can handle incomplete training data. The Voting EM (Expectation-Maximization) algorithm is one such algorithm [1]. Bayesian updating is another. It is only applicable to variables which do not have parents in the graph.

Each variable in a Bayes net is associated with a Conditional Probability Table (CPT) that gives the probability of each state of the variable conditional on each combination of states of its parent variables. When a variable has no parents, its CPT is simply an assignment of prior

probabilities to its states. The Voting EM algorithm allows us to learn the CPTs of the observed variables as well as other variables such as explicitly represented error variables. It is also robust to fundamental changes in the model. When we do not explicitly model errors and do not assume the observed variable's modeled behavior is correct, we must use the Voting EM algorithm to learn the CPT for the observed variable. Since these variables have parents, we can not apply Bayesian updating.

Bayesian updating is an online, incremental approach which adjusts a CPT using Bayes' rule. For our models, it can learn the distributions for error modes such as bias or effective sample size when these are explicitly represented in the model (and do not depend on other variables). It is incapable of learning the CPTs of observed variables since these have parent variables.

In order to test our approach, we must have a BN model which is to accurately represent the behavior of the system. We also create a base model which is loosely related to the "true" model. The base model will have less informative CPTs and may include additional error variables. Learning is applied to the base model with training cases generated from the ground truth model as shown in Figure 13. This adjusts the CPTs for the observed variables or error variables (such as bias and effective sample size).

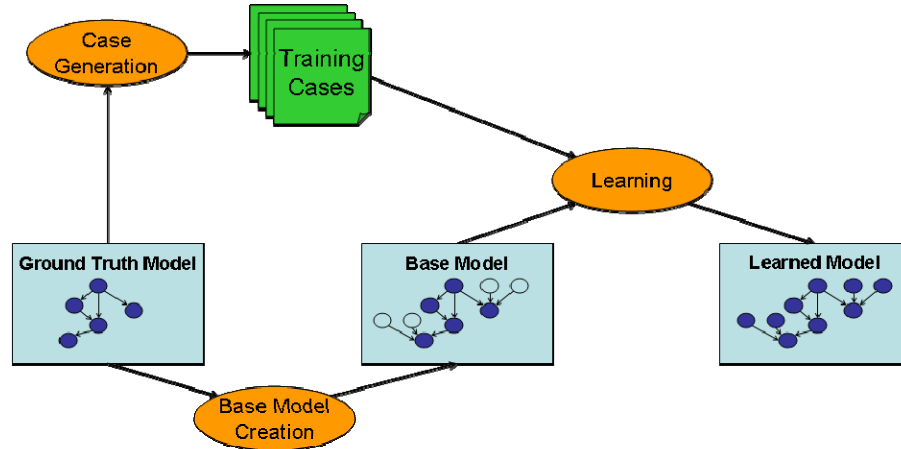


Figure 13. The Learning Process.

We now discuss how we test the performance of the learned models. To do so, we must have test cases simulated by the ground truth BN. These cases may have some evidence nodes unfilled. We define the parameter observability to be the probability that any given observation will have a value. Thus observability influences how many of the values will be missing in the cases. We now compute raw scores for the ground truth, base, and learned models using these cases as shown in Figure 14. We describe the scoring algorithm in the next paragraph.

We measure success in terms of how accurate the inference is to the correct value of the hypothesis of interest. For this example, the hypothesis of interest is whether the facility has power. If the result of updating the model were either a definite "Power On" or "Power Off" conclusion, then we could count up the number of cases in which the correct answer was obtained and take the proportion of right answers out of all answers as the average accuracy of

the inference. However, the result of updating is typically not a definite value, but rather probabilities for its values. We can give “partial credit” in such a situation. If, for example, updating on the evidence results in a probability of 0.9 for “Facility Power = Power On” *and* the facility has power, then we say that the answer is 90% accurate or accurate to degree 0.9. If, instead, the facility has no power, then the accuracy of the answer is $1 - 0.9$ or 0.1. In this way, by assigning partial degrees of accuracy to updates, we can sum up these partial degrees of accuracy and arrive at an average partial accuracy. This provides us with an average raw score for a model.

To score the learned model, we compute a relative score as shown in Figure 14 – namely the progress from the base model to the learned model with respect to the ground truth model. Thus if the base model has a raw score of 0.6, the learned model has a raw score of 0.77, and the ground truth model has a raw score of 0.8, then the score of the learned model is $\frac{0.77 - 0.6}{0.8 - 0.6} = 0.85$.

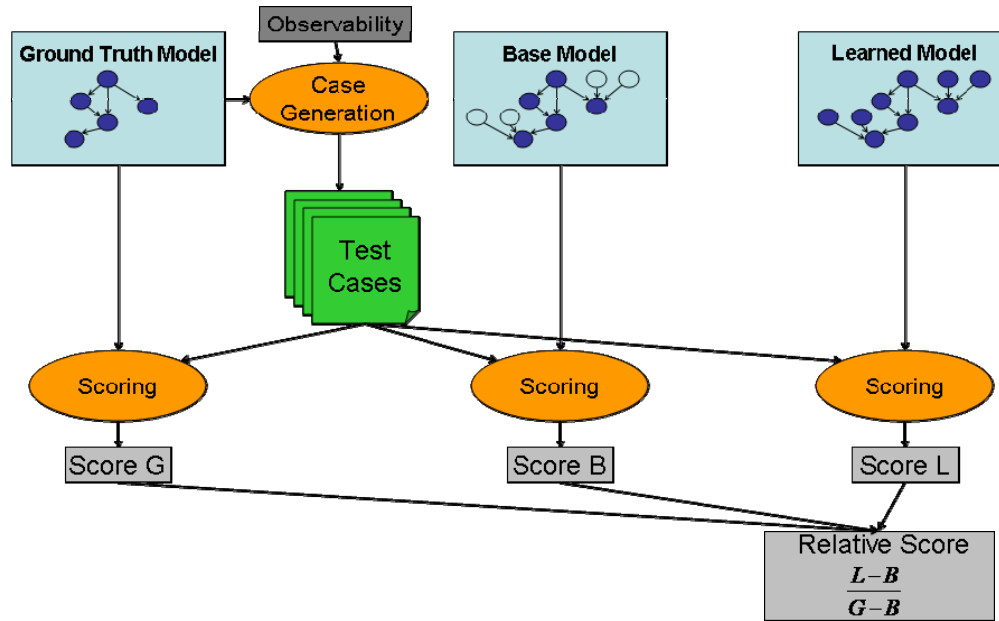


Figure 14. Testing (and Scoring) the Learned Model.

Note that when we are applying (as opposed to testing) our data validation methodology, we construct a model using expert knowledge (if available) that captures the system as accurately as we can. We then perform learning on it if training data is available. The result will be a learned model which we can use for data validation. If independent test data is available, we can compare the raw scores of the original model and the learned model to measure the improved predictive ability of the learned model.

6.2 RESULTS

To test the validation frameworks, we created several BNs with different types of error models. Some had a variable for reliability where as others had effective sample size and bias. We extend our model it by adding error models and additional reports. For the extended model, we add an additional (independent) damage report for the facility. Also we add three additional reports for each of the sensors that detect whether the facility has power. The four readings from each sensor are gathered consecutively in a short span of time. We also added error models. Each sensor at each time period may be functional or non-functional – this is captured by the “sensor mode” error variables. Each of the damage reports has a reliability variable which indicates the accuracy of the source. For the complete model, see Figure 15.

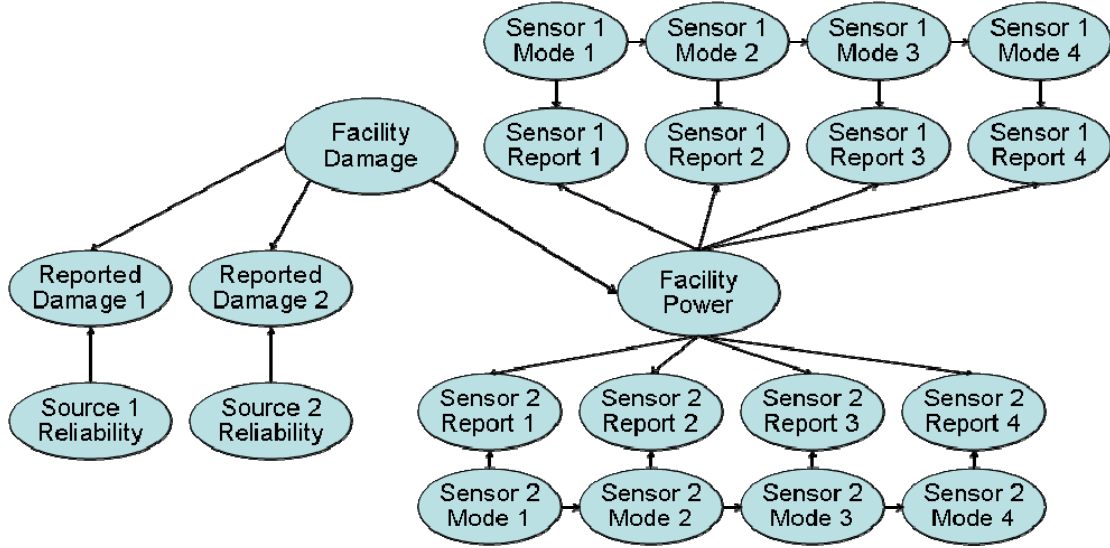


Figure 15. Model for Testing Data Validation Framework.

For the network depicted in Figure 15, we generated a number of training cases with certain percentages of the observations missing. We learned three models – one using 100 training cases, one using 500 training cases, and another using 1000 training cases. We show the performance of the Voting EM algorithm in Figure 16. Note that even when the observability is low, the algorithm performs extremely well given enough training cases.

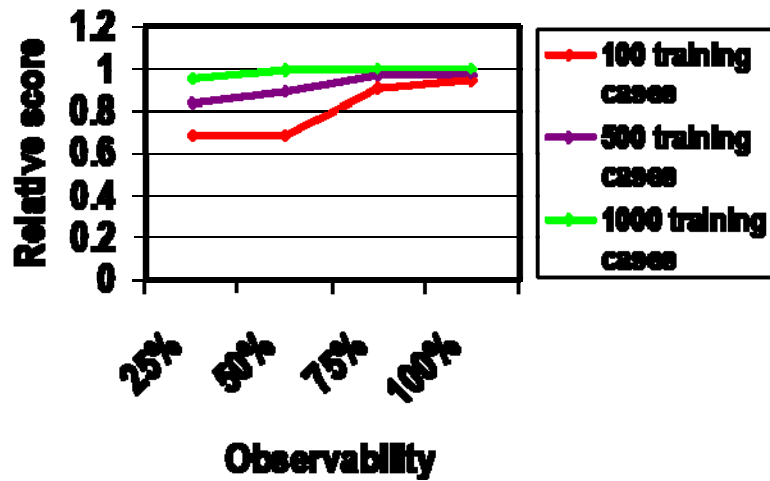


Figure 16. Performance of Voting EM Algorithm on Test Model.

Using a model very similar to the one in Figure 15, we compared the scores using Bayesian updating and the Voting EM algorithm. The training cases were 50% observable. In this test, Bayesian updating does very well – it has approximately the same predictive ability as the true model after about 100 training cases. It clearly outperforms the Voting EM algorithm. However, one must bear in mind that the Voting EM algorithm can also learn the CPTs for observation nodes whereas Bayesian updating is unable to do this. In addition, the raw score for the base model was 0.7 and the raw score for the ground truth model was 0.77. In other words, the difference between the ground truth and base models was not too large to begin with.

Table 2. Performance of Adaptive Learning vs. Bayesian Updating

Training Set Size	Improvement from baseline	
	Voting EM	Bayesian Updating
20	24%	58%
100	26%	91%
500	28%	100%
1000	38%	95%

In some of our tests, the learned models have scores of 100% or greater. We believe this is because most of the generated cases have few or no observation errors. For these cases, the learned model outperforms the true model. However, for those cases with more observation errors, the true model outperforms the learned model.

To test bias and effective sample size learning, we took the model in Figure 15 and introduced bias in the system reports (in the true model). The bias was introduced within the CPTs for the system reports and not with explicit bias variables. We then created a base model with bias and effective sample size variables which were learned using the Voting EM algorithm. We show the scores in Table 3. These results are very promising. Note that the model learned from only 100 cases performs very well.

Table 3. Performance of Model with Bias and Effective Sample Size Error Models

Training Set Size	Observability			
	25%	50%	75%	100%
100	73%	90%	90%	93%
500	99%	100%	99%	99%
1000	102%	101%	101%	100%

7 I2AT INTERFACE

The I2AT interface was designed with two major points in mind. The first was to clearly display the information generated by the I2AT application, and the second was to make data entry simple.

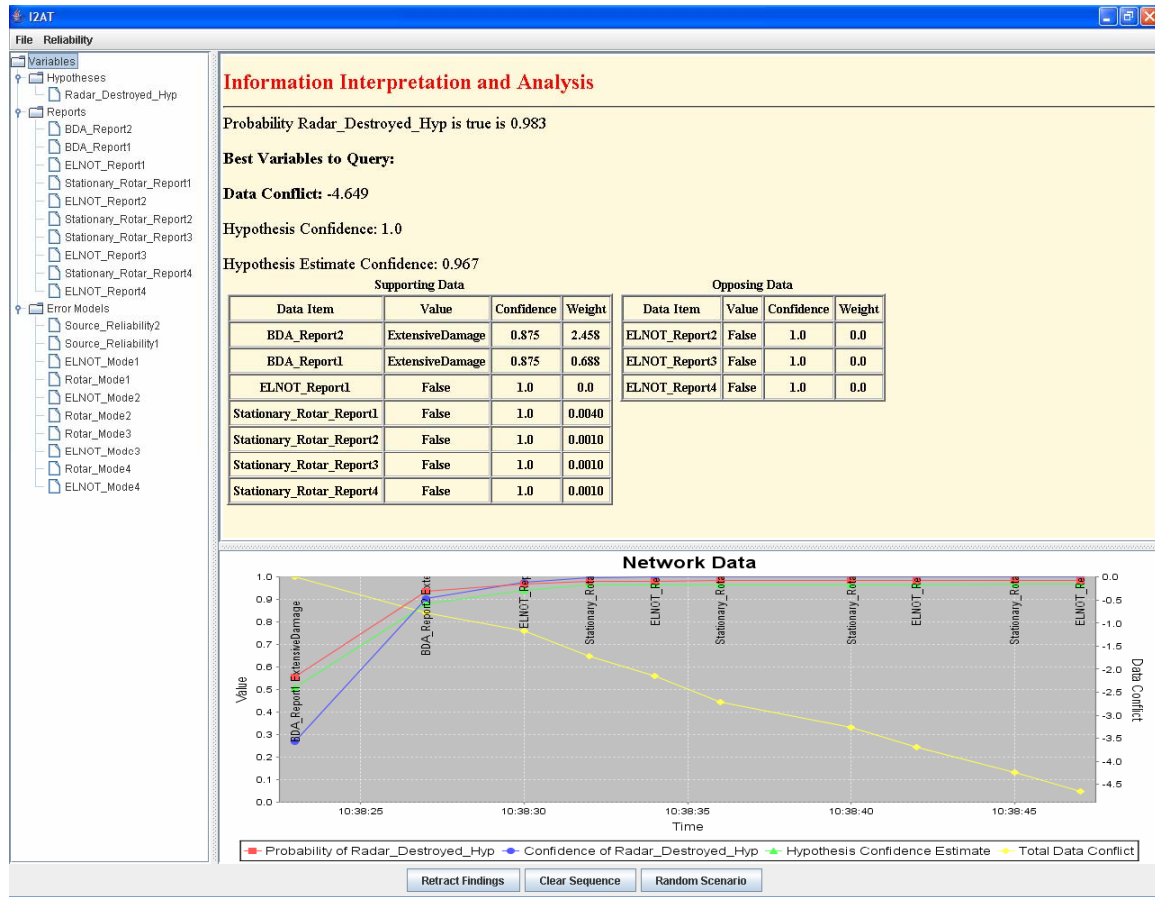


Figure 17. I2AT User Interface

To achieve these goals, the main I2AT window is separated into three major components as in Figure 17. The frame on the left is the data input area. The top right frame is the data display area and the bottom right area is the data graph.

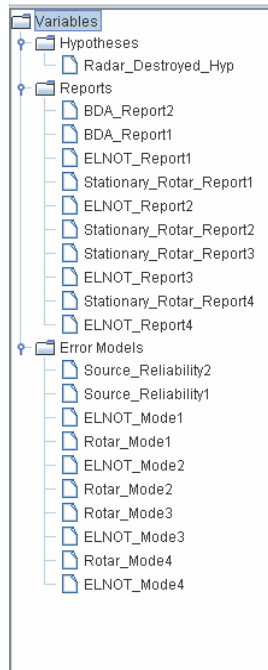


Figure 18. Data Input Frame

The data input frame (Figure 18) presents nodes to the user based on node type. The node types displayed are hypothesis, report, and error nodes. When a node is selected, a new window pops up, based on the node type. For hypothesis nodes, the hypothesis selection window is opened (Figure 19). In this window, the user can choose to make the selected node the active hypothesis. All data for the model will then be recalculated with respect to the new hypothesis.

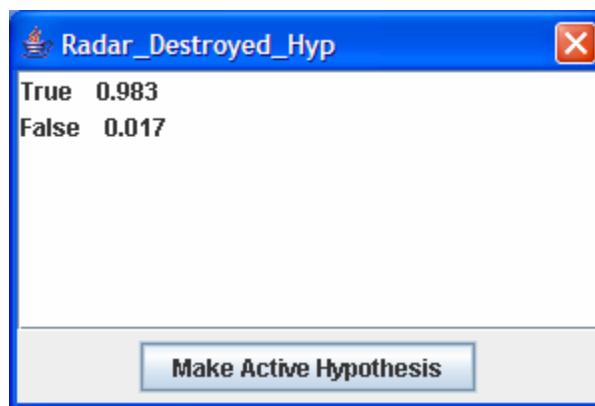
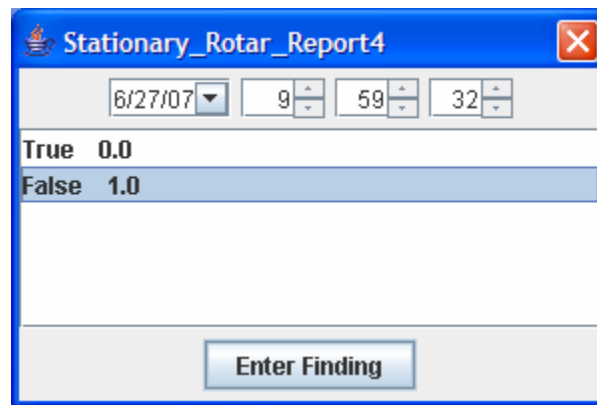


Figure 19. Hypothesis Selection Window

For a report or error node, the data input window is opened (Figure 20). This window has multiple values that can be set. The user is required to select a node state before it can be entered

into the model. Optionally, the user can also set the date and time this value was received. If no value is entered into the date and time fields, the current time is used.



The window titled "Stationary_Rotar_Report4" contains a date field set to "6/27/07" and three time fields set to "9", "59", and "32". Below these fields, there are two rows: "True 0.0" and "False 1.0", with the "False 1.0" row highlighted. At the bottom is a button labeled "Enter Finding".

Figure 20. Value Input Window

Information Interpretation and Analysis							
Probability Radar_Destroyed_Hyp is true is 0.0040							
Best Variables to Query:							
Most informative query: Stationary_Rotar_Report2							
Best utility node: Stationary_Rotar_Report2							
Data Conflict: 0.595							
Hypothesis Confidence: 0.993							
Hypothesis Estimate Confidence: 0.993							
Supporting Data				Opposing Data			
Data Item	Value	Confidence	Weight	Data Item	Value	Confidence	Weight
BDA_Report1	ModerateDamage	0.0050	0.336	BDA_Report2	NoDamage	0.931	-3.012
ELNOT_Report2	False	0.021	1.667	ELNOT_Report1	True	0.979	-1.616
				Stationary_Rotar_Report1	True	0.979	-2.648

Figure 21. Data Display Frame

The data display frame (Figure 21) is used to disseminate information about the model. There are two major types of data. The first are model wide values calculated with respect to the currently selected hypothesis. The second is node specific data. The node specific data is segregated into tables based on the node's effect on the hypothesis. Nodes in the Supporting Data column support the current hypothesis, while nodes in the Opposing Data column do not. The information in this frame is updated anytime new data is entered into the model.

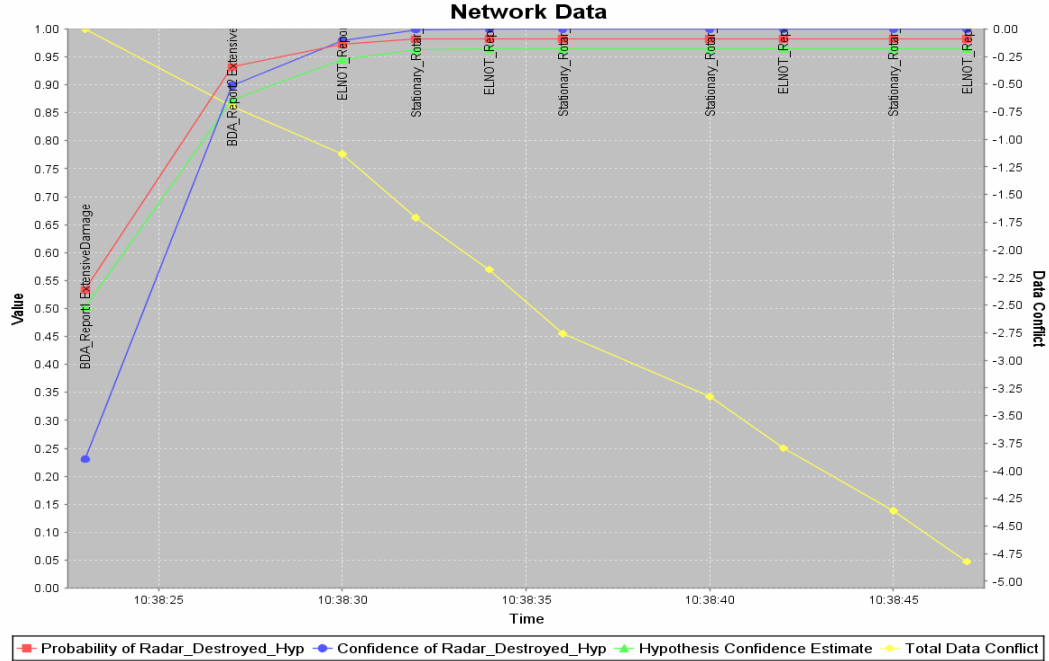


Figure 22. Data Graph

The graph frame (Figure 22) is used to graph the changes in model wide values over time. It is redrawn when new information is input. The graph displays the change in model wide calculations over time.

8 REASONING ACROSS SCENARIOS

One problem with using a single scenario model is that no information determined by the model is preserved. While report values, and the probability of the hypothesis do not have meaning outside of the singular scenario, the derived values of error nodes can have meaning. For example, in our radar attack model, if we believe that a sensor is malfunctioning in one scenario, this belief would be useful in the next scenario involving the same sensor. (For example, in a different scenario in which a different target is attacked but the same sensors from a previous attack were employed.) A set of scenarios like this, with the same report sources, is called a sequence. The use of sequences creates more accurate hypothesis beliefs.

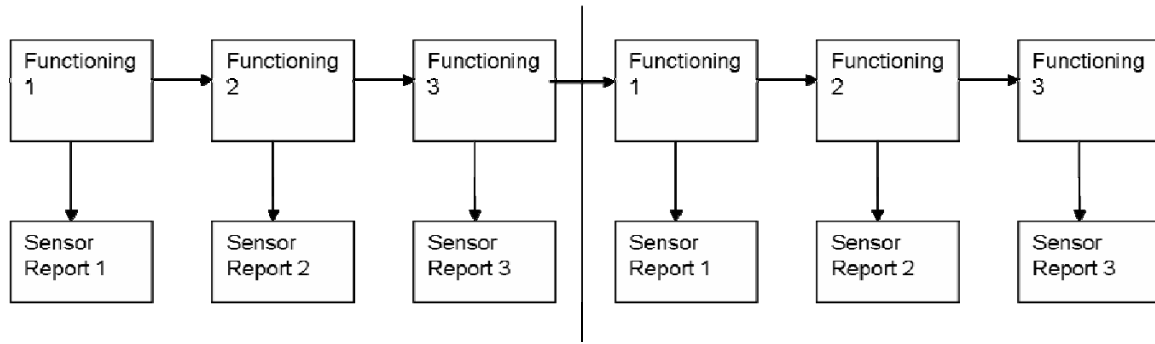


Figure 23. Large Model Approach.

A simple approach to modeling a sequence would be to build networks with multiple scenarios (Figure 23). This would link error values between scenarios. This link would allow error nodes to take into account past and future error beliefs, and generate more accurate predictions. However, this approach has two major problems. First, the models used to do this would need to be large enough and complex enough to model all possible sequences. Second, due to their larger size, I2AT would perform significantly worse.

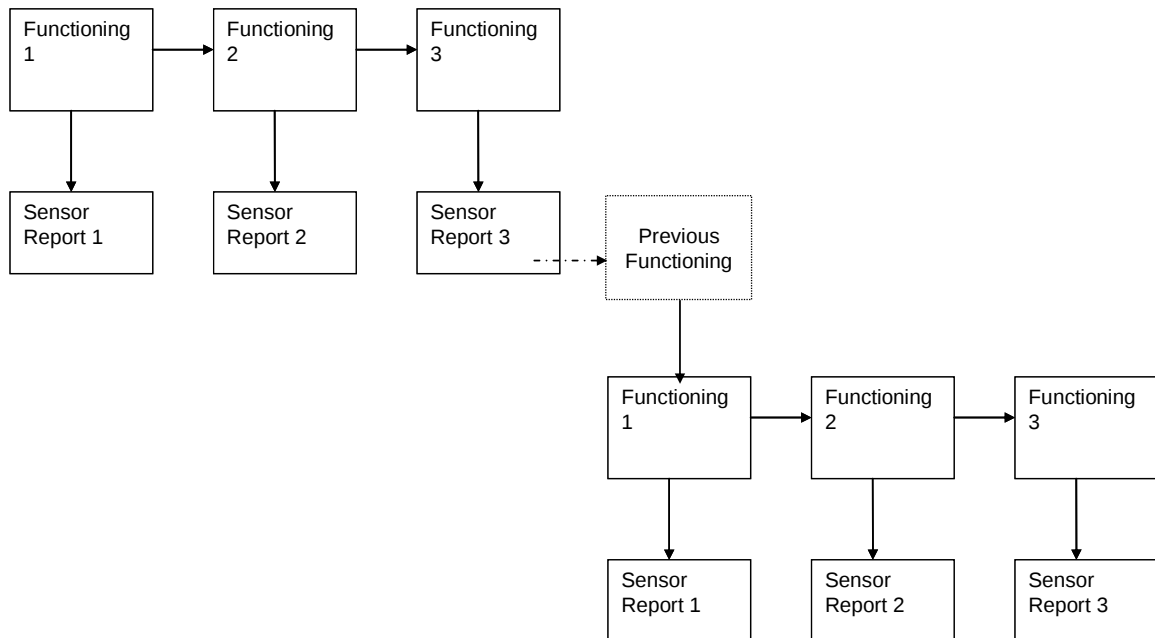


Figure 24. I2AT Approach

To get around these limitations, I2AT uses a different approach. Instead of using one large network, I2AT models can have special nodes that are used to propagate past error values into the new scenario (Figure 24). This propagation allows for the historical error values to affect the

current scenario while preserving the smaller network sizes. The downside is that past scenarios are not affected by newer information.

Another feature of this approach is information can be shared between related models. If two models have the same error nodes and nodesets, the information copied into a new scenario will be the most recent, independent of which model generated the information. This can allow for collaborative models to be created.

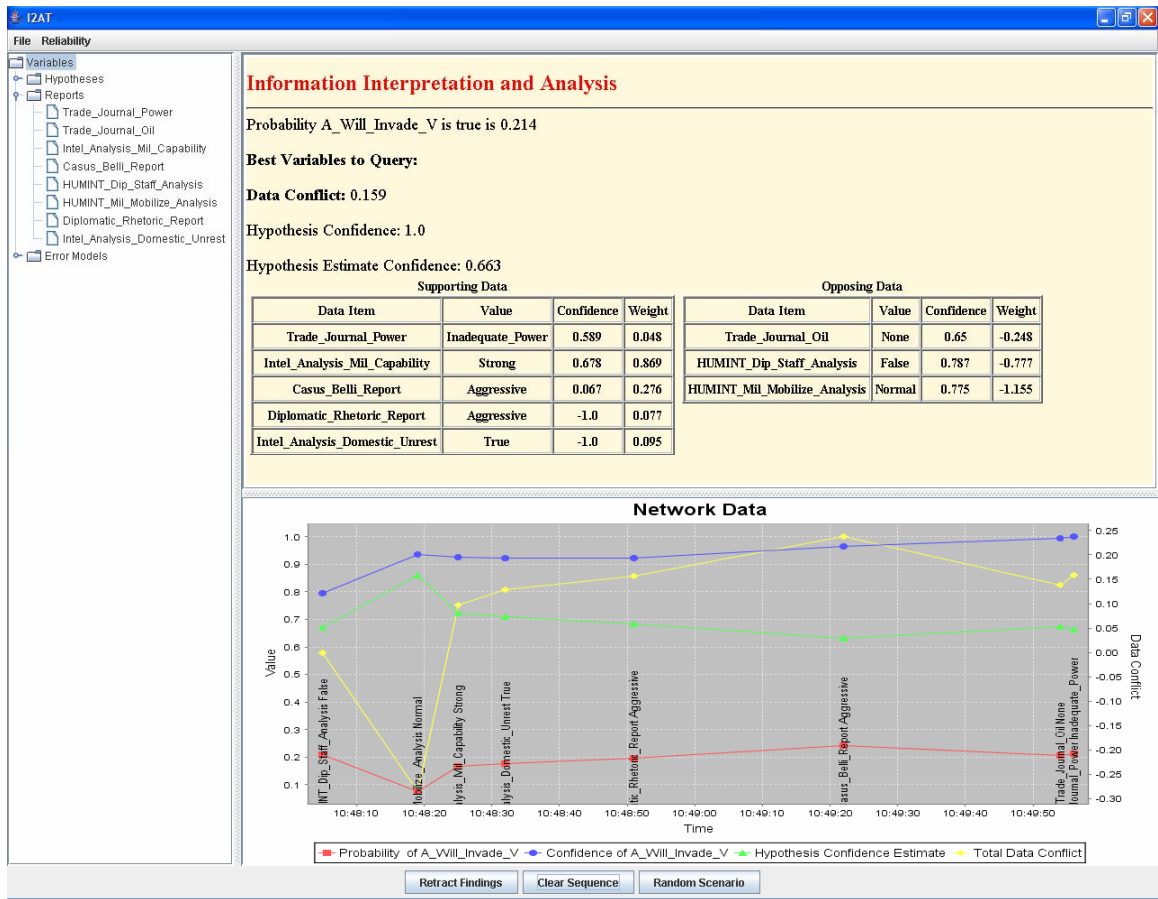


Figure 25. Pre Sequence Threat of War

The effects of this system can be observed in I2AT by creating a scenario, and seeing the difference in values if it is part of a scenario sequence. Figure 25 shows a scenario using the Threat of War model, without any error history considered. The probability of the hypothesis being true is only 0.214.

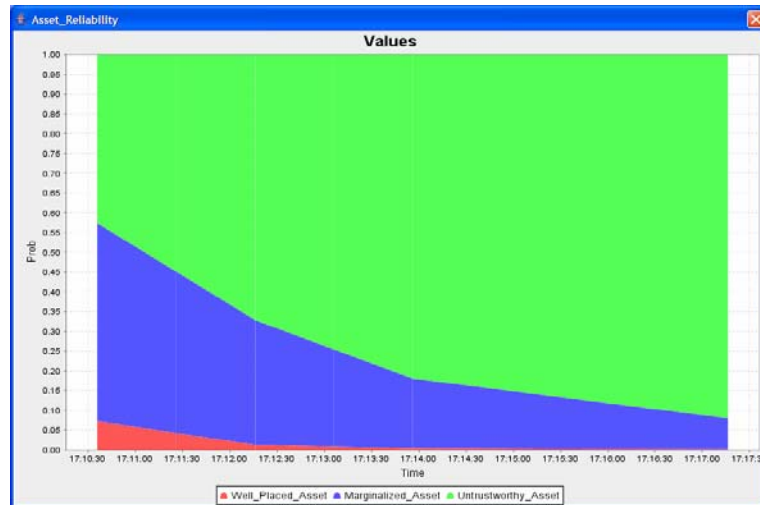


Figure 26. Asset Reliability

If other past scenarios are considered, our belief in the reliability of some sources may change. Figure 26 displays how evidence from past scenarios has led I2AT to believe that the Asset Reliability is untrustworthy.

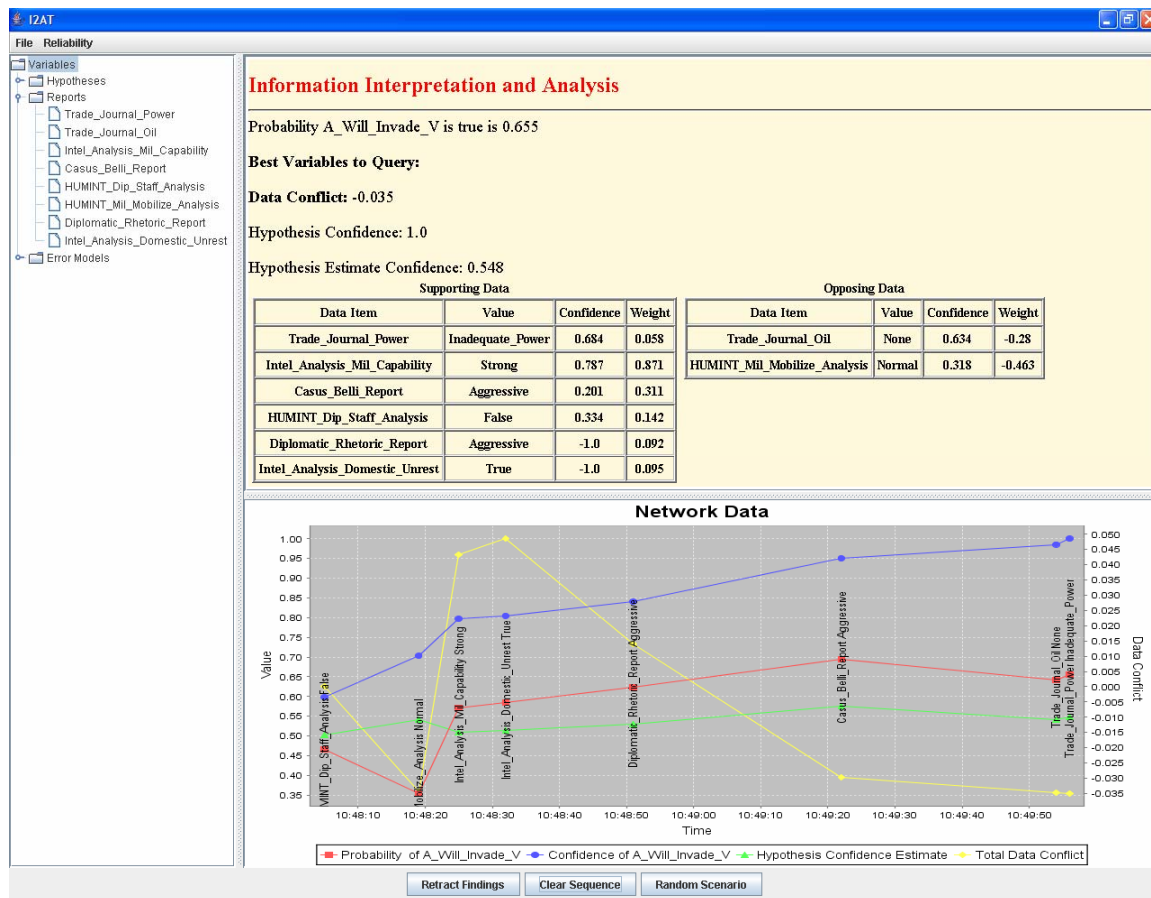


Figure 27. Threat of War scenario with error propagation.

If the scenario used in Figure 25 is reloaded with our asset reliability lowered, I2AT determines a different set of beliefs. The hypotheses value in Figure 27 has changed to 0.655, despite identical report values from the first scenario. This shows how I2AT, using past error values, can evaluate data more accurately, while still minimizing model sizes.

9 APPLYING I2AT TO WEB-BASED DATA VALIDATION

9.1 OPEN SOURCE INFORMATION GATHERING

Previous work has focused on data validation through the modeling of information sources. This is a “push” model of data validation, since we assume that the information sources are known and have provided the interpretation system with their input. Another approach to data validation is a “pull” approach in which potentially new information sources are sought out in order to assess data provided by some known source. Increasingly, web-based information sources are providing value open-source information to the intelligence community, due to the

explosion of information on the Web in the form of blogs, traditional news outlets, and propaganda and disinformation sites (which can actually provide useful information if their bias and motivations are known). In November 2005, the Open Source Intelligence Center was opened at CIA headquarters with the mission to exploit such non-traditional information sources.

The potential value of open source information is limited, however, by the vast sea of irrelevant information in which it swims, making it impossible to manually extract and examine all potentially relevant items of information. We need a system that is able to automatically detect relevant sources of information and use those sources to assess the validity of data. To that end, we have prototyped a system, whimsically called “Doctor Knowledge” that searches the Web for information about a particular claim and uses the results to assess the validity of the claim.

9.2 WEB-BASED DATA VALIDATION

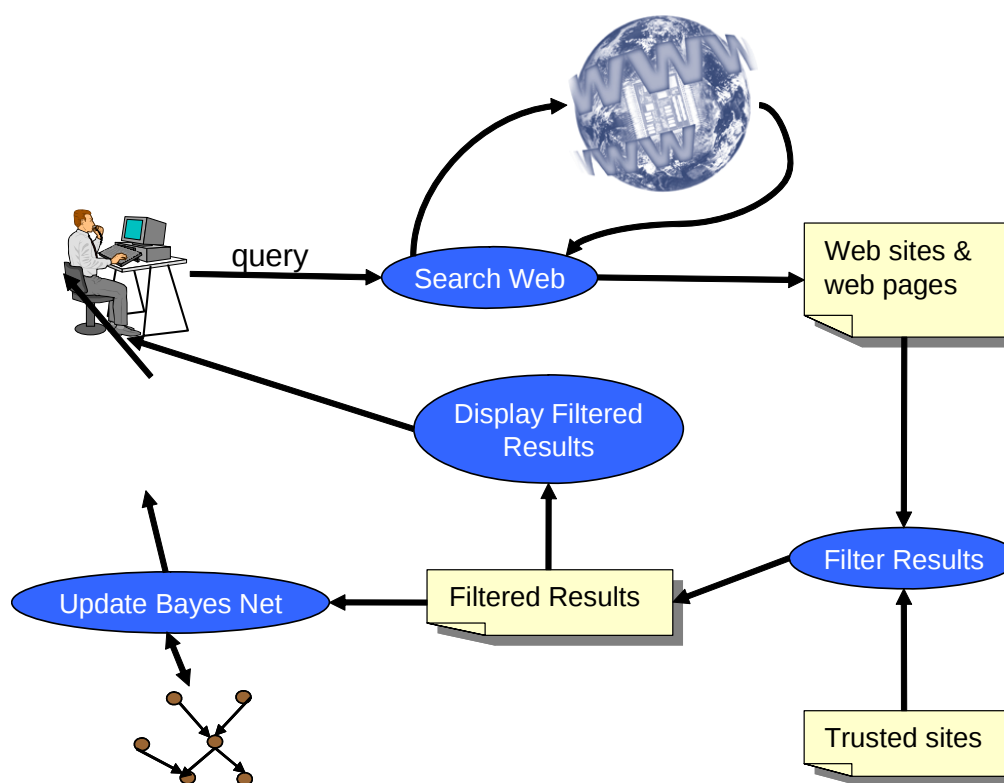


Figure 28. System architecture for web-based data validation.

Figure 28 shows the architecture for Doctor Knowledge. The user enters a query concerning some data item which he wants validated (e.g. “Osama bin Laden has been captured”). The query is entered into some standard web search engine such as Google and results are returned. A list of modeled sites is maintained by the system and the pages returned by the query that are not associated with any of the modeled sites are filtered out. The remaining results are analyzed and used to update a Bayesian network. The Bayesian network models what sort of information is found on different types of sites and how reliable those sites are. For example, simply

knowing that a claim is discussed on a reliable hoax debunking website is evidence that the claim is false. In some cases, the actual content of the website can be extracted, providing more specific information about whether it is asserting the claim to be true or false, or is taking no position.

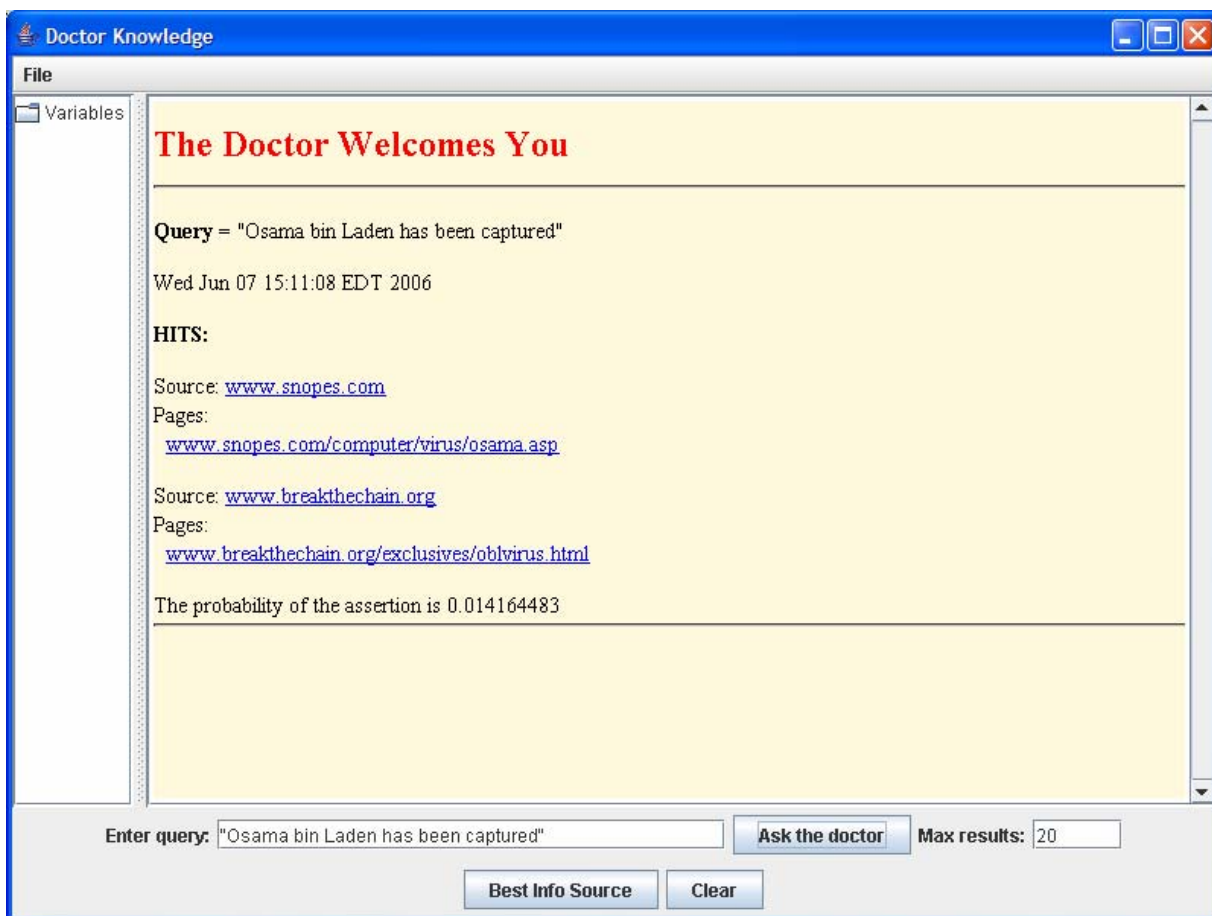


Figure 29. Web pages returned by Dr. Knowledge.

As shown in Figure 29, the web pages returned are displayed as clickable links so that the user can see their contents if he or she so desired. Based on the types of websites found, Doctor Knowledge makes a preliminary estimate of the probability of the entered claim. In this case, because all the returned links are for “hoax-busting” websites, the claim is deemed very improbable. If the user does not wish to investigate all the returned links, the user can ask for a recommendation for a subset of links that are likely to be most informative about the claim in question. This is determined from a computation of the expected information gain for each returned page.

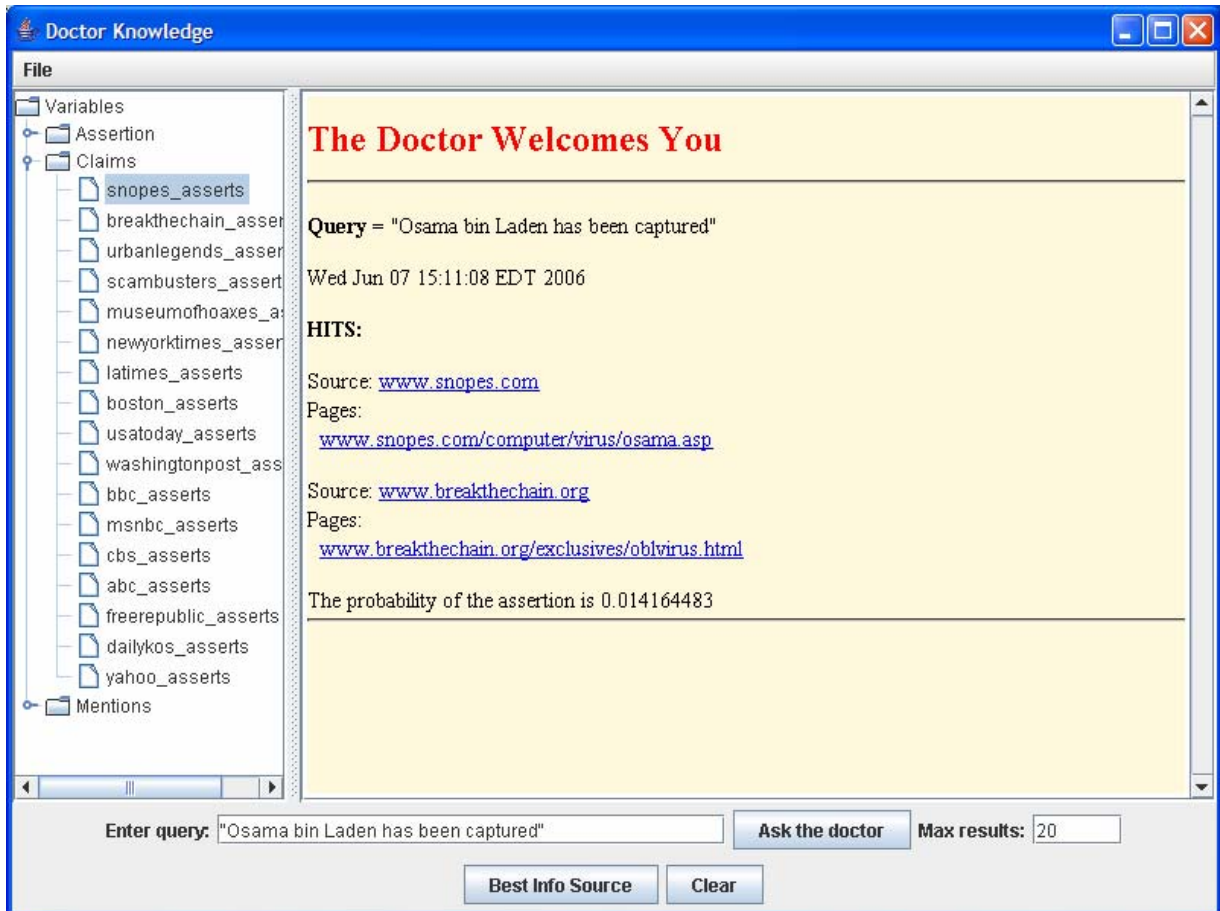
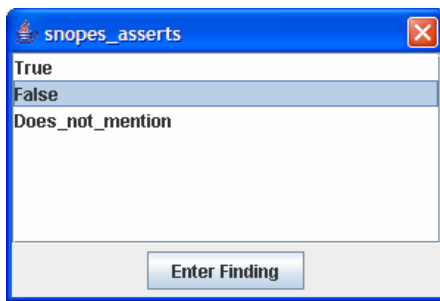


Figure 30. Entering a variable value in Dr. Knowledge.

After extracting the content of particular web pages, the user can enter the information learned directly into the Bayes net. To do so, he or she selects the appropriate variable from the list on the left and a window pops up in which the user can enter the value corresponding to the learned information. In this case, we suppose that the user has gone to the Snopes site and determined that it does indeed assert that the claim in question is false.

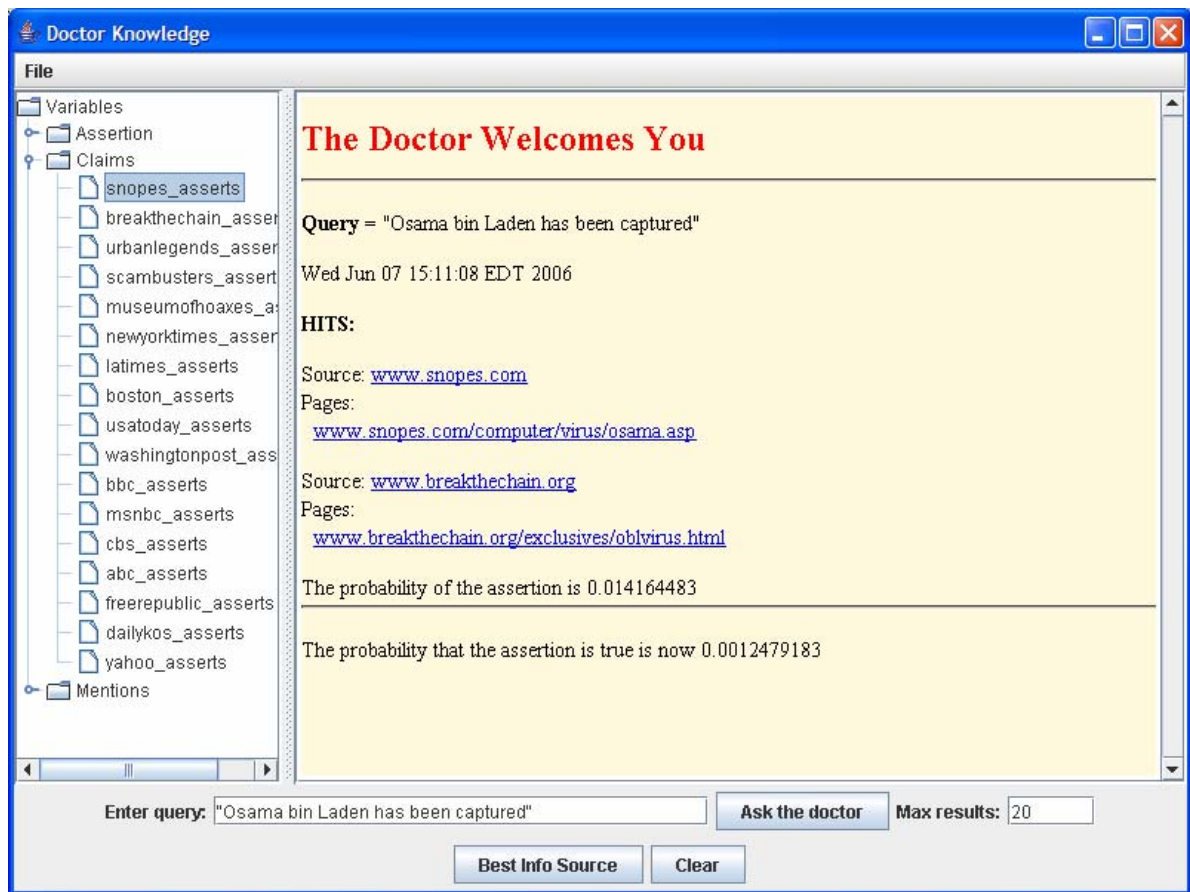


Figure 31. Updated probability of claim.

The value entered causes the Bayes net to update its probabilities and the new probability of the claim is displayed.

In order to achieve a reasonable level of performance, we trained the Bayesian network model. To start this process, we identified the type of each website (for example, "urban legend debunker" or "news"). This information was used to establish preliminary parameter estimates. These determined the degree to which a search hit for a website suggests a claim is true.

We scoured the web for various claims -- some true, some false. These were fed into Netica's training algorithm to fine tune the parameters of the model. Since the size of the training set was relatively small, we employed a smoothing technique to reduce over fitting. To this end, we first identified clusters of similar probability tables in the network. The smoothing then took each element in a cluster and moved it closer to the cluster center. Finally, we validated the resulting model to ensure that the resulting model was acceptable.

9.3 LIMITATIONS OF DR. KNOWLEDGE

Our experiment with web-based data validation was fruitful in that it revealed significant potential for automating data validation and information collection over the web. At the same time, it revealed serious limitations to what can be done with current technology. The most serious limitation, not surprisingly, is the limited ability to extract *content* from web pages. Dr. Knowledge has limited content extraction capabilities, confined to a small number of web sites containing known structure. For the most part, Dr. Knowledge does not rely on the content of a web site to determine the validity of an assertion but only on the nature of the web site (e.g. hoax exposing web sites generally – but not always – discuss claims that are false).

To realize the full potential of Dr. Knowledge, it would have to be integrated with a system for content extraction. Moreover, the content would have to be translated into assignments over variables in a Bayes net. Unfortunately, current content extraction technology is still very much in the research stage and translation of content into formal representations is far in the future except for structured sources whose semantics is known.

Another limitation of Dr. Knowledge stems from the querying mechanism, which searches for exact matches to the string entered. Ideally, we would like to search for web pages that are relevant to the *content* of the query. This is the content extraction problem from another angle and so its solution awaits advances in content extraction technology.

10 MODELING COLLABORATIVE INTELLIGENCE ANALYSIS

10.1 WMD EXAMPLE

The Netica file *WMD_Nonprolif.dne* contains a Bayes Network causal graph model of a WMD nonproliferation intelligence analysis problem. The node outline is shown in Figure 32 and the full model with probabilities is shown in Figure 33. The model examines two questions:

1. Given reports from various intelligence sources, what is the probability that the adversary has an active military nuclear program? Thus one hypothesis node is called *Has_Mil_Nuclear_Program*, with a value of true or false.
2. Given reports from various intelligence sources, what is the probability that the adversary has a functional nuclear weapon? The other hypothesis node is called *Has_Nuclear_Weapon*, with a value of true or false.

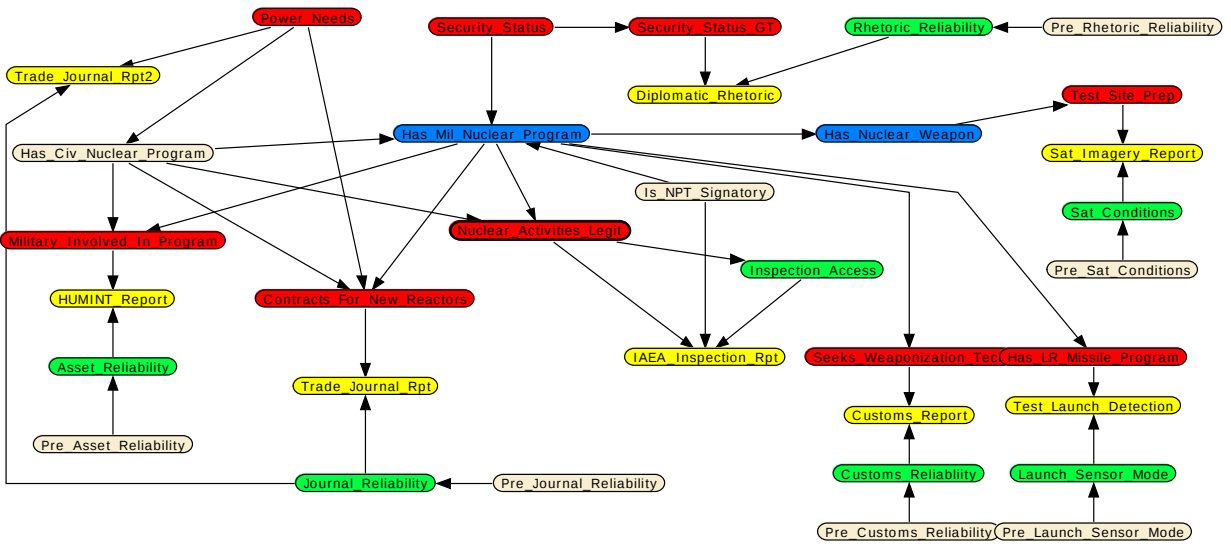


Figure 32. Nodes of WMD Nonproliferation Bayes net model

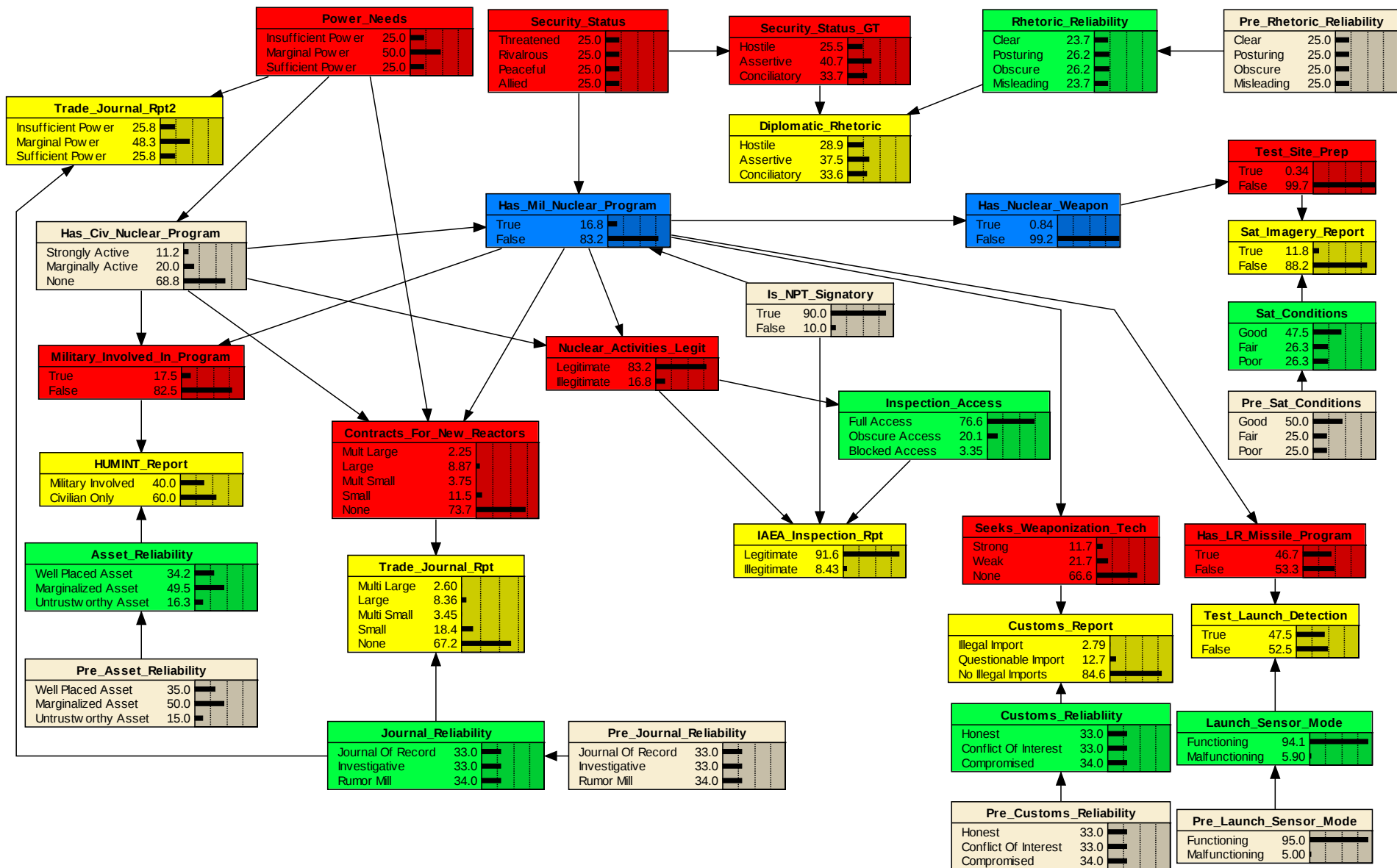


Figure 33. States and Probabilities of WMD Nonproliferation Bayes net

model

Hypothesis of Military Nuclear Program

Many pieces of ground truth information influence our belief that Red has a military nuclear program, but may only indicate that Red has a legitimate civilian nuclear program. Having a civilian nuclear program, however, does greatly increase the likelihood that Red also seeks military nuclear technology. These factors are:

- The current unmet power consumption needs of Red's economy (node *Power_Needs*). Red's economy may have **Insufficient Power** (power needs greatly overwhelm generation capacity), **Marginal Power** (power needs are growing about the same rate as Red's generation capacity growth), or **Sufficient Power** (power generation capacity exceeds economic needs). If Red has insufficient or marginal power, it is more likely that Red will have a legitimate civil nuclear power program and be securing contracts for the construction of nuclear power plants (node *Contracts_For_New_Reactors*).
- Red's current geopolitical security status (node *Security_Status*). Red may believe, in increasing order of security, that Red is **Threatened** (one or more hostile nations), **Rivalrous** (one or more rivals for regional or global status), **Peaceful**, or **Allied** (depends on US for a collective security arrangement). Decreased security status increases the likelihood that Red seeks military nuclear technology.

The four states in this node are mapped to three states for the ground truth node (*Security_Status_GT*). This is required because the report node *Diplomatic_Rhetoric* contains only three states, and the report and ground truth nodes must match in number of states. The third state (**Conciliatory**) in both nodes (*Security_Status_GT* and *Diplomatic_Rhetoric*) is presumed to map equally to the last two states (**Peaceful**, and **Allied**) in node *Security_Status*.

- Red's military involvement in the countries nuclear activities (node *Military_Involved_In_Program*). If Red is using the military to provide security for nuclear facilities, using military bases to house nuclear research, or has some military personnel involved in nuclear program oversight, Red may be concealing military interests in nuclear technology.
- Red's contracts with international companies to build nuclear power reactors. Depending on the scale of these contracts relative to Red's power needs, this could indicate interest in military nuclear applications.
- The legitimacy of Red's nuclear activities (node *Nuclear_Activities_Legit*). Red's acquisition of nuclear technology may or may not be in accordance with relevant export control laws and commitments to the Nuclear Nonproliferation Treaty (NPT).
- Red's seeking advanced technologies for weaponizing nuclear materials, such as uranium enrichment via centrifuge, or plutonium reprocessing (node *Seeks_Weaponization_Tech*). Red may be strongly, weakly, or not seeking such technologies.
- The existence of a long range missile program (node *Has_LR_Missile_Program*). Since long range missiles are a crucial means for effective delivery of nuclear weapons, the existence of such a program may indicate that Red's nuclear interests are of a military nature.

Since we do not have access to the ground truth, we rely on intel reports to help us infer our hypothesis:

- Red's power needs and public contracts for nuclear reactors are reported by nuclear energy industry trade journals (node *Trade_Journal_Rpt* and *Trade_Journal_Rpt2*). Trade journals have varying reputations: highly reliable Journals of Record, more speculative Investigative journals, and less reliable journals that tend to circulate rumors to scoop competitors.
- Red's subjective estimates of security status are inferred from analysis of his diplomatic rhetoric (node *Diplomatic_Rhetoric*) – whether it is **Hostile**, **Assertive**, or **Conciliatory**. Red is more likely to respond with more hostile or assertive rhetoric if insecure, and more conciliatory if secure. However, Red's rhetorical posture influences how we might understand his rhetoric – Red may be **Clear**, **Posturing**, **Obscure**, or deliberately **Misleading**.
- In this model, we have a HUMINT asset in place that can inform us about military involvement with nuclear programs (node *HUMINT_Report*). This asset's information is more or less reliable depending on whether he is well-placed, marginalized, or simply untrustworthy.
- We detect possibly illegitimate nuclear activities through an IAEA Inspection Report. However, the degree to which we believe this report depends on the level of access we believe the inspection team had to Red's facilities. In fact, the more Red has to hide, the more likely it is that inspection teams will not have full access, which decreases our confidence in this report.
- We detect Red's attempts to secure weaponization technologies through Customs Reports, which may reveal a clearly illegal import, a questionable import, or legal imports only. However, customs officials may have varying degrees of reliability, depending on whether they are Honest, have a Conflict of Interest, or are actually compromised by Red.
- Evidence of a long range missile program is provided by a Test Launch Detection. The reliability of the detection is determined by the state of our launch detection sensor suite, which may be functioning or Malfunctioning.

Hypothesis of Producing a Nuclear Weapon

If Red has a military nuclear program, Red may have produced a nuclear weapon (in this model, Red cannot have a nuclear weapon without a military nuclear program).

One piece of ground truth information influences our belief in whether Red has a nuclear weapon – whether Red has constructed a test site whether a nuclear weapon can be exploded (node *Test_Site_Prep*). If Red has a nuclear weapon, Red may construct a test site, but will not construct a test site if he does not have a weapon. We do not have access to the ground truth, so we rely on reports from satellite imagery to detect the existence of a test site (node *Sat_Imagery_Report*). In our model, the satellite imagery may be ambiguous depending on the satellite viewing conditions, such as weather. The node *Sat_Conditions* has values Good, Fair, or Poor, in order of decreasing reliability of whether a satellite image contains an identifiable test site.

10.2 THREAT OF WAR EXAMPLE

The Netica file *Threat_of_War.dne* contains a Bayes Network causal graph model of an intelligence analysis problem where analysts are trying to determine the likelihood that **Country A** (Aggressor) will go to war with **Country V** (Victim). The model is shown in simplified form in and in full form displaying states and probabilities in .

The causal network is primarily intended to be used to answer the questions Given reports from various intelligence sources, what is the probability that A will invade V? Thus one hypothesis node is called *A_Will_Invade_V*, with a value of true or false. Several ground truth facts might cause A to invade V:

- A's military capability relative to V (node *A_Military_Capability*), which may be stronger than V, have parity with V, or be weaker than V. The stronger A's military relative to V, the more likely it is that A might invade V.
- A might have an ethnic motive to invade V (node *A_Ethnic_Motive*). If so, two states influence whether A might have an ethnic motive. First, V may or may not have an ethnic minority (node *V_Has_Ethnic_Minority*). Second, there might be domestic unrest in V (node *V_Has_Domestic_Unrest*). This unrest may be Open, Submerged, or nonexistent. If both of these factors hold, then A might invade V to support a related ethnic minority uprising in V.

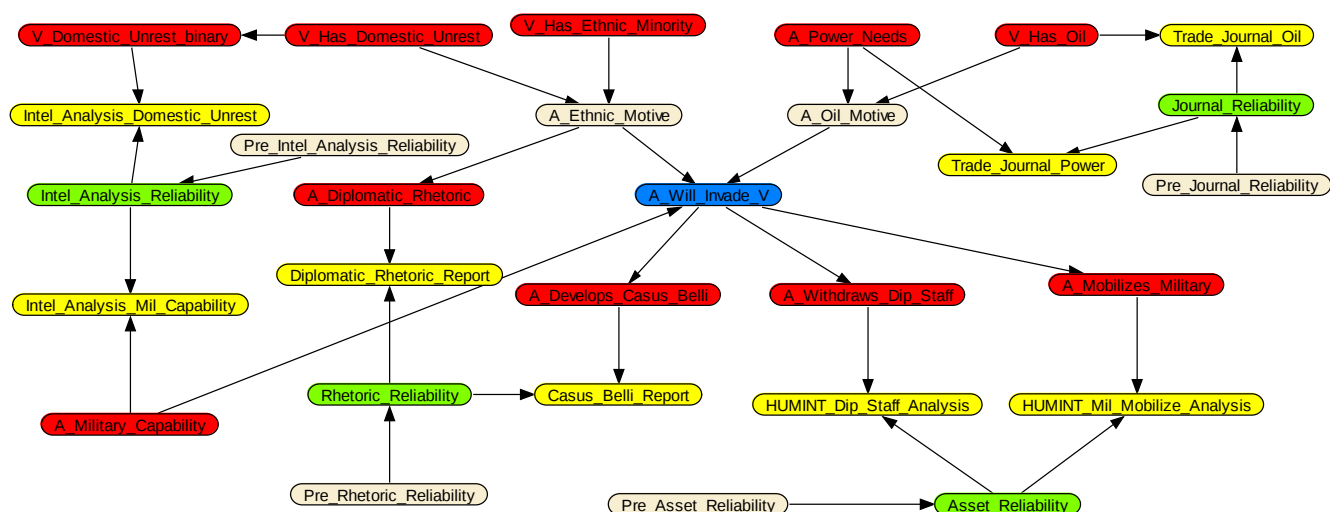


Figure 34. Threat of War Bayes net model

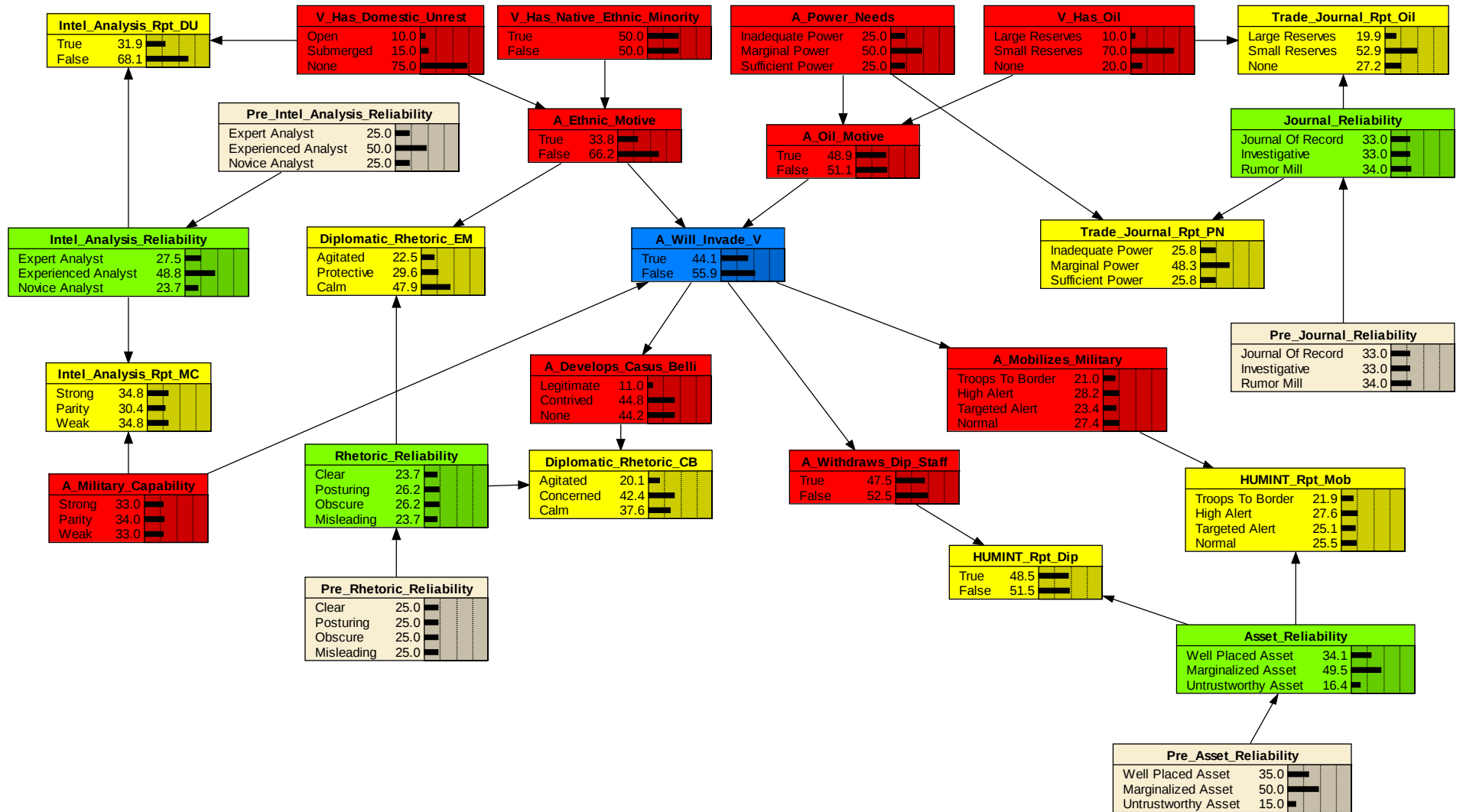


Figure 35. Threat of War Bayes net model with states and probabilities

The ground truth node *V_Has_Domestic_Unrest* is mapped to a binary version of the same node, called *V_Domestic_Unrest_binary* in order to match the number of node states with the report from intelligence analysis (node *Intel_Analysis_Domestic_Unrest*) which contains two states: true, and false.

- A might have an oil motive to invade V (node *A_Oil_Motive*). If so, two states influence whether A might have an oil motive. First, A's economy might have unmet power consumption needs. Thus node *A_Power_Needs* can take on values of insufficient power generation capacity (power needs greatly overmatch generation), marginal capacity (power needs are growing about the same rate as Red's economic growth), or sufficient capacity (power generation has enough slack to accommodate growth). Further, V might have oil reserves, thus node *V_Has_Oil* may take on the values Large Reserves, Small Reserves, or None. If A's need for oil is matched by V's available reserves, then A might invade V to appropriate V's oil resources.

Many pieces of ground truth information further influence our belief that A is preparing to invade V. These facts are:

- A generates diplomatic rhetoric in the node *A_Diplomatic_Rhetoric*. This rhetoric is reflective of their ethnic motive to invade, greater motive leads to greater degrees of rhetoric (aggressive).
- A develops a *casus belli*. Through diplomatic statements, A is laying groundwork to legitimize an imminent invasion of V to the world community or to its population. The node *A_Develops_Casus_Belli* may have the values Legitimate, Contrived, or None.
- A withdraws diplomatic staff from its embassy and other posts in V (node *A-Withdraws_Dip_Staff*). This act is usually taken as a sign of imminent conflict.
- A mobilizes its military forces (node *A_Mobilizes_Military*). There are varying degrees of mobilization or readiness level for A. In decreasing order, these are moving troops to V's border, placing all forces on high alert but not moving them, placing some special forces on alert (Targeted Alert), or normal readiness.

Since we do not have access to the ground truth, we rely on intel reports to help us infer our hypothesis:

- Reports that might inform us about an oil motive for A derive from trade journal reports. A's power needs and V's oil reserves are reported publicly by energy industry trade journals (nodes *Trade_Journal_Power* and *Trade_Journal_Oil*). Trade journals have varying reputations: highly reliable Journals of Record, more speculative Investigative journals, and less reliable journals that tend to circulate rumors to scoop competitors. (Note: This report source is intended to be the same in the WMD and War models to demonstrate propagation of source reliability across intel efforts.)
- Reports that might inform us about an ethnic motive for A, or in general reveal A's development of a casus belli, derive from analysis of A's diplomatic rhetoric. A's diplomatic rhetoric (node *Diplomatic_Rhetoric_Report*) about ethnic concerns in V can

be Aggressive, Moderate, or Calm, in decreasing order of concern. A's *casus belli* rhetoric (node *Casus_Belli_Report*) can be Aggressive, Moderate, or Calm. A is more likely to be planning to invade the higher the level of concern in the rhetoric. However, A's rhetorical posture influences how we might understand his rhetoric – he may be Clear, Posturing, Obscure, or deliberately Misleading. (Note: This report source is intended to be the same in the WMD and War models to demonstrate propagation of source reliability across intel efforts.)

- In this model, we have a HUMINT asset in place that can inform us about military mobilization (node *HUMINT_Mil_Mobilize_Analysis*) and withdrawal of diplomatic staff (node *HUMINT_Dip_Staff_Analysis*). This asset's information is more or less reliable depending on whether he is well-placed, marginalized, or simply untrustworthy. (Note: This report source is intended to be the same in the WMD and War models to demonstrate propagation of source reliability across intel efforts.)
- We assess A's military capabilities relative to V and V's state of domestic unrest through intelligence analysis (nodes *Intel_Analysis_Mil_Capability* and *Intel_Analysis_Domestic_Unrest*). The reliability of intel conclusions is directly related to the experience level of the intel analysts. Intel analysts can be Novice, Experienced, or Expert.

11 EXTENSIONS TO I2AT

This section describes ways of extending I2AT and integrating it with other decision support systems

11.1 CAUSAL REASONING WITH JCAT

Dr. John Lemmer of AFRL Rome has managed the development of a tool named JCAT, which is an evolution of a Bayesian Network analysis tool developed at Alphatech (now BAE AIT). BAE maintains a codebase for this original implementation, called the Operational Assessment Tool (OAT).

JCAT/OAT enables construction of node-link networks that display probabilities of node states given the causal factors influencing that node. OAT uses nodes as actions (or effects) that take on certain values (or states), and links among the nodes to signify causal relationships. These links contain probabilities and a sign: positive (enhancing) or negative (inhibiting). For example, Node1 might cause Node2 through positive link with influence probability $p=0.6$.

JCAT/OAT can run these models over time, giving a temporal representation of the evolution of probabilities of the nodes. Links that cause or inhibit nodes can also contain delays (once started, delayed in influencing the effect node for a period of time), and links can be persistent (once started, continue acting for a period of time).

The major advantage of JCAT/OAT is the support given to a user (modeler) in constructing probabilistic causal models. The user does not enter Conditional Probability Tables (CPTs) as they would when specifying Bayesian networks. Instead the user enters probability values for nodes and influence values for links. The software essentially generates the CPTs from this information. In addition, JCAT/OAT contains mechanisms that allow a temporal representation, including stepping through time (simulation), Monte Carlo sampling at a time point, and delays and persistence that allow causes to maintain (or delay) the influence for some time period.

JCAT/OAT Extensions to I2AT

Two extensions to I2AT would support better modeling using existing tools – JCAT/OAT.

- Probabilistic Causal Modeling (**PCM**) tool to support causal model building, and export of resulting model for I2AT data analysis
- I2AT data and evidence statistics exported into the **PCM** tool

The first extension to leverage I2AT assessment and analysis would be the development or modification of a tool to enable modelers to build **causal models** in the style of JCAT/OAT, and have these save down to Bayesian networks with full CPTs that are accessible by the I2AT algorithms. One motivation for providing this capability is to enable easier data and evidence entry through causal model building, rather than through Bayes net model building with the requirement of entering full Conditional Probability Tables. Another motivation is the lack of data validation tools within OAT itself. Some versions of JCAT/OAT have contained Bayes evidence comparison tools; however they have not covered the range of capabilities of the I2AT algorithms.

The second extension would enable I2AT to export data back into the **PCM** tool to update the current causal probabilities or states with better values determined by I2AT.

11.2 JOINT DATA/MODEL VALIDATION

I2AT provides data validation and data interpretation and analysis. Its primary focus is determining whether data used in the model are correct. However, the model itself might be wrong. The model is a modeler's hypothesis of the situation (evidence of relationships among nodes) given his/her knowledge of the world or knowledge gained from other sources (SMEs / other experts / open source / closed source, etc.) and is subject to mistakes in the source of model information and in translation of that information into a formal model.

So another I2AT extension would be a support tool that helped a modeler develop an accurate, proper, "valid" model of the situation. This extension would be supported by Bayesian inference algorithms that helped produce statistics for likelihood of certain structures over other structures.

For example, we might have two structures that contradicted each other, or at least were fundamentally different in their assumed causality, and we wish to know which structure is holding up best to the situation, given various evidence we find about the real world.

$A \rightarrow B$ or $C \rightarrow B$?
 $A \rightarrow C \rightarrow B$ or $A \rightarrow D \rightarrow B$?

Which structure is correct? Can we use evidence and confidence on evidence to select one or another of these assumed (hypothesized) structures?

Or, given $A \rightarrow B \leftarrow C$, are both A and C really influential to B? Or can one of them be shown to be unimportant, or simply not true (not connected)?

The CPTs of a Bayesian network state assumption about the influence of one or more variables on another variable. So our current I2AT algorithms might easily answer this. What we haven't talked about is a way to inform the modeler so that they might drop a node or edge from the model. As models get refined and evolve, modelers remove as well as add structure. If C is initially believed to be influential (say $p = 0.3$) but is shown to be hardly there at all ($p = 0.0000001$), then the modeler might wish to remove C from the model.

Another possibility for research is into automated searches (optimize/sensitivity testing) that look for nodes which have very little influence over outcomes, even if their probabilities of occurrence are large or change radically over time.

11.3 TIME-DEPENDENT VALIDATION SCHEMES

The scenarios considered in I2AT have so far concerned relatively static situations – e.g. whether a particular radar is operating, whether a country has a nuclear weapons program. One extension of I2AT would be to look at data about ground truth states that vary over time and exploit temporal correlations to determine data/model validity.

Data from different nodes that are temporally highly-correlated indicate (or allow inference) that the nodes are somehow connected. Likewise, time-series data displays actual node behavior (evidence) and if a parent node is changing value or state (with a high degree of certainty), then child nodes would be expected to change also. Time-series or other correlations might show that structure or connection between nodes is likely, or unlikely, and thereby provide additional evidence for the values in the CPTs (or the causal probability links in JCAT/OAT).

11.4 OPEN SOURCE INFORMATION GATHERING

Our experiment with web-based data validation (“Dr. Knowledge”) showed the need for some form of content extraction or analysis in order to achieve satisfactory levels of performance. Augmenting Dr. Knowledge with content extraction and analysis capabilities, even if imperfect, would significantly increase the accuracy of its results. We could, for example, integrate the open source document classification and information extraction tool MALLET [7] into Dr. Knowledge. MALLET could facilitate Dr. Knowledge’s reasoning in several ways. First, it could help to determine the relevancy of a web site to the given query; second, it could help classify a previously unknown web site; and third, it could perform some limited content extraction that could indicate the *attitude* of a web site document to the query.

Another approach to extending Dr. Knowledge would be to make use of the type of information collected by the “Dark Web” project at the University of Arizona [8]. This project collects statistics concerning terrorist web sites. The statistics collected include clustering of sites based on which sites link to other sites, some content extraction, classification of sites in terms of level

of hate or advocacy of violence, and ideology. This information could be used by Dr. Knowledge to determine the relevance and reliability of a web site for a particular query as well as whether claims made on multiple web sites are made independently or are possibly simply passed from one web site to another. All these factors affect the way in which evidence about web site relevance and content should impact the probability of a claim.

12 REFERENCES

- [1] Cohen, I., Bronstein, A., Cozman, F. *Online Learning of Bayesian Network Parameters*, Hewlett-Packard Technical Report HPL-2001-55, 2001. Available at <http://www.hpl.hp.co.uk/techreports/2001/HPL-2001-55R1.pdf>.
- [2] Cowell, R. Dawid, A., Lauritzen, S., Spiegelhalter, D. *Probabilistic Networks and Expert Systems*, Springer-Verlag, 1999.
- [3] Jensen, F. *Bayesian Networks and Decision Graphs*, Springer-Verlag, 2001.
- [4] Parsons, S. "Current approaches to handling imperfect information in data and knowledge bases", IEEE Transactions on Knowledge and Data Engineering, 8(3), pp. 353-372, 1996.
- [5] Pearl, J. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan Kaufmann, 1988.
- [6] Schum, D. *The Evidential Foundations of Probabilistic Reasoning*, Northwestern, 2001.
- [7] MALLET. Information available at http://mallet.cs.umass.edu/index.php/Main_Page.
- [8] Dark Web Project, University of Arizona, AI Lab: <http://ai.arizona.edu/research/terror/index.htm>.

13 APPENDIX: MODEL CONSTRAINTS

13.1 NODE TYPE CONSTRAINTS

Nodes in I2AT models should consist of several different types. Each of the types is essential to the operation of the application, and need to be present for calculations to work.

Hypothesis nodes are nodes that contain a true/false hypothesis about the state of the model. For example, on the default FEBO model, the hypothesis is that the radar is destroyed. While I2AT supports multiple hypothesis nodes in a single model, all hypothesis nodes must only contain two values, true and false.

Ground Truth nodes are nodes that represent actual real world values. These nodes are not presented in the I2AT application, as the model assumes that these values are not known to the observer, but only inferred from reports.

Report nodes represent observations or other information the model needs to consider. I2AT uses these values to infer the values of the ground truth nodes.

An error node represents a measure of the accuracy of the associated report node. It can be a reliability value, a sensor state value, or any other measure of performance.

The model can have non typed nodes; however these nodes are hidden and not accessible from the I2AT application. Non typed nodes can be used to add additional complexity to the model.

13.2 METADATA REQUIREMENTS

Additional information about the nodes is needed by I2AT to successfully operate. These data items are stored as UserData in the Netica model. Each UserData item has a label and a value. We will discuss each label I2AT needs, the values it can take, and if the label is required or optional.

The first label is “Type”. This label is used to identify the node type. The recognized types are “gt” for ground truth nodes, “hyp” for hypothesis nodes, “report” for report nodes, and “error” for error nodes. If there is no Type data for a node, it is untyped.

The second label is “NodeSet”. A NodeSet is a collection of related nodes. All nodes in the NodeSet are required to have the same set of state values. All error nodes must belong to a nodeset.

The third label is “Pre”. This label is used to mark a node as a posterior error node. These special nodes are used to propagate error information from past scenarios. The node marked with pre must belong to the same NodeSet as the error node it links with. A model does not require these nodes to work, but will not propagate information without them.

The final label recognized by I2AT is “Order”. This label is used to identify the order of a node inside a NodeSet group. This is an optional field, and is only used if the nodes have an inherent order in which their values can be determined.

13.3 STRUCTURAL REQUIREMENTS

A model needs to conform to three structural requirements in order to be successfully used by I2AT. The first two requirements relate to the link structure of the model. The first is a report node needs a link from one ground state node. The second is error nodes need to link to at least one report node. The third requirement relates to the states of a report node and its parent ground truth node. For the calculation of individual data confidence values, the states of the report node and its parent ground truth node must be the same, and in the same order. This one to one mapping is required by the mechanism used to calculate data confidence of reported values. If this requirement is not met, the values calculated for data confidence will be meaningless.