Making Sense of VAST data

Summer Adams & Ashok K. Goel

Artificial Intelligence Laboratory & Southeastern Regional Visual Analytics Center
Division of Interactive and Intelligent Computing
Georgia Institute of Technology
Atlanta, GA 30332, USA
{summer@cc.gatech.edu, goel@cc.gatech.edu}

Abstract: We view the task of sensemaking in intelligence as that of abducing a story whose plot explains the current data and makes verifiable predictions about the future and the past. We have developed a computational system, called STAB, that abduces stories from data. The story plots in STAB are represented as processes with goals and states, and organized in an abstraction hierarchy. STAB analyzes the VAST dataset generated by PNNL. This dataset pertains to normal and typical activities, as well as illegal and unethical activities, in a fictitious town in the United States. Given the VAST data incrementally, STAB retrieves and invokes multiple story plots as explanatory hypotheses and generates expectations about future data. It uses supporting and contradicting evidence to build justifications for its final conclusions.

Keywords: Intelligence analysis, sensemaking, abduction, explanation.

1. Introduction

Making sense of data in intelligence analysis [Krizan 1999, Heuer 1999, Thomas and Cook 2005] generally involves the tasks of recognizing and characterizing a problem based on some initial evidence about an event or activity, generating multiple explanatory hypotheses based on the evidence, collecting and assimilating additional data, evaluating the multiple explanatory hypotheses, and selecting the most plausible hypothesis. The sensemaking task is very complex because of the constantly evolving, and often unreliable and conflicting nature, of the data. The evolving nature of data implies a need for ongoing monitoring and continual generation and evaluation of hypotheses so that new evidence can be accounted for as it arrives and the most confident explanation can be produced at any given time.

Pirolli and Card [2005] describe an information-processing model of intelligence analysis based on a cognitive task analysis of human analysts as they did their jobs. They have identified two main, overlapping loops in the analyst's problem solving, a foraging loop and a sensemaking loop. Figure 1 describes these two loops. The foraging loop involves finding the right data sources; searching and filtering the information; and extracting the information [Pirolli and Card 1999]. The sensemaking loop [Pirolli and Card 2005] involves iterative development of a conceptualization (a *hypothesis*) from stored *schema* that best fits the evidence, and the presentation of the knowledge product that results from this conceptualization.

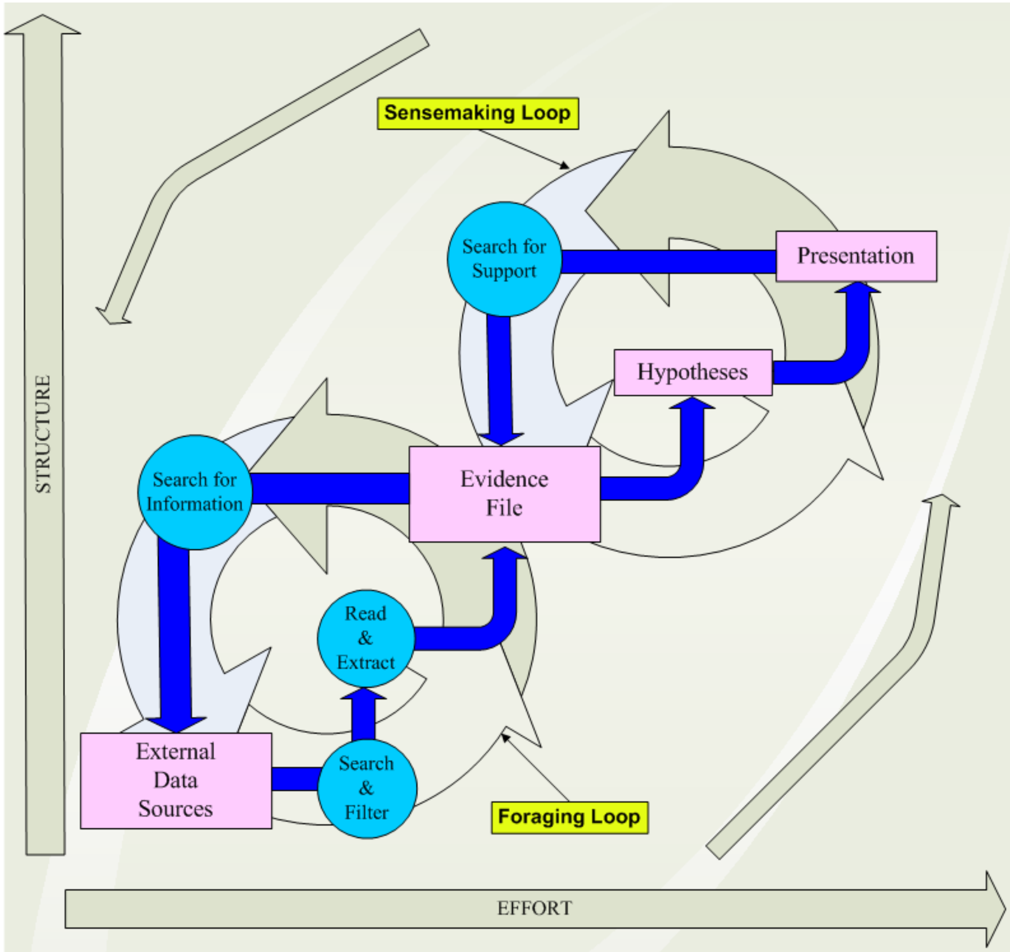

Figure 1. The Sensemaking and Dataforaging Loops in Intelligence Analysis (Adapted from Pirolli and Card 2005)

Pirolli and Card, however, do not describe the content and the structure of the schemas in their sensemaking loop, or the process by which specific schemas are conceptualized as hypotheses. We have developed a computer system, called STAB (for STory ABduction) that seeks to answer precisely these questions.

We view the task of sensemaking as that of abducing a story whose plot explains the current data and makes verifiable predictions about the future and the past. The abduction task in general takes as input a set of data, and gives as output an explanatory hypothesis that best explains the data [Josephson and Josephson 1994]. In STAB, the explanatory hypotheses are in the form of stories. The stories are represented as processes with goals and states, and organized in an abstraction hierarchy. STAB analyzes the VAST (for Visual Analytics Science and Technology) dataset generated by the Pacific Northwest National Laboratory (http://conferences.computer.org/vast/vast2006/). This dataset pertains to normal and typical activities, as well as illegal and unethical activities, in a fictitious town in the United States. Given the VAST data incrementally, STAB retrieves and invokes multiple stories as explanatory hypotheses and generates expectations about future data.

2. Stories and Sensemaking

STAB contains a library of generic, skeletal story *plots* relevant to the VAST domain. We handcrafted this library of plots. Figure 2 illustrates the plot of a simple story in STAB's library, which is composed of several smaller plots. The main plot (in the middle of the figure) is to Rob a Store, which has several steps to it: Go to Store, Break into Store, Take Money. This plot has the goal of Have Money, given the initial state of Not Have Money (top of figure). Each of the steps in this plot can potentially be done using multiple methods. For example, the step of Break into Store can be done by Entering through a Window or Entering through a Door (bottom of figure). Each of these methods in turn is a process consisting of multiple steps.

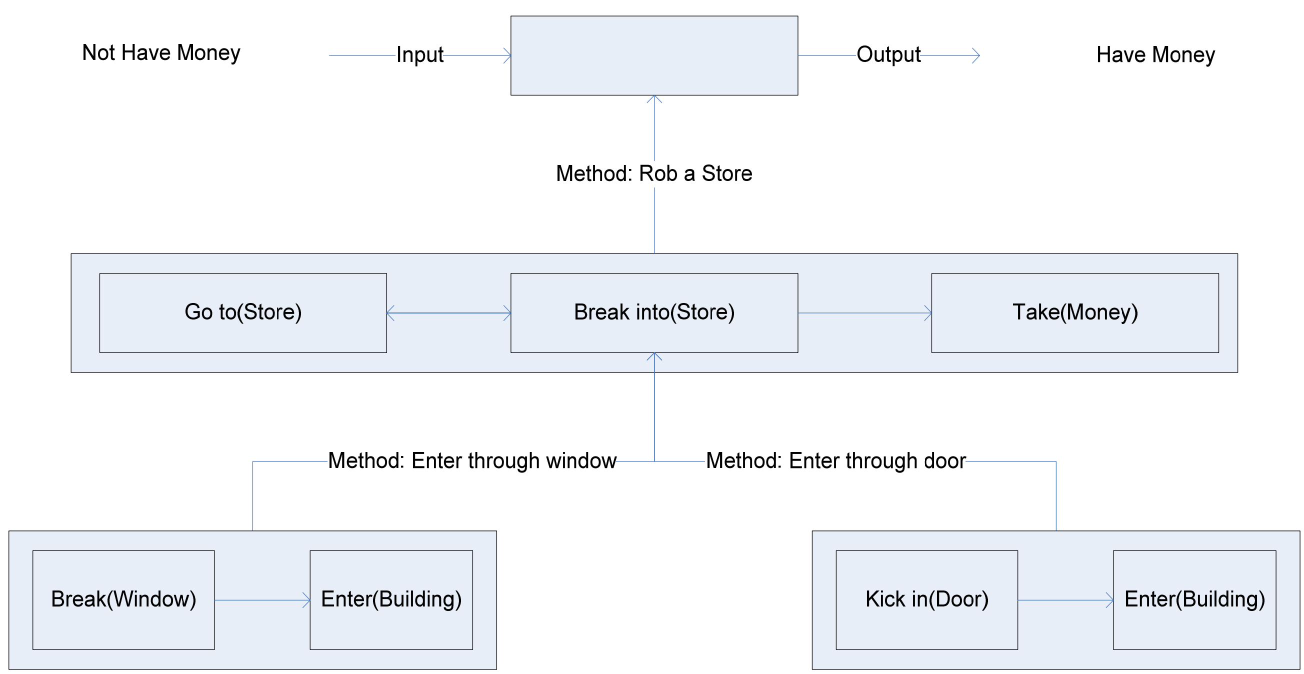

Figure 2: The Content and Structure of a Story in STAB

A story plot in STAB is like a *script* [Schank and Abelson 1977]. However, while a script represents a sequence of actions, a plot in STAB also represents the goals of the action sequences and the knowledge states produced by each action. These knowledge states play an important role in the retrieval of plots relevant to a given data. STAB uses the TMKL (Task-Method-Knowledge Language) knowledge representation language

[Murdock and Goel 2001] to represent story plots. TMKL has the expressive power of Hierarchical Task Networks [Sacerdoti 1977].

3 From Story Plots to Explanatory Hypotheses

Figure 3 shows the high-level architecture of STAB. First, The Evidence Collector collects the input events in an Evidence File in chronological order. Table 1 shows a sample of the input events to STAB for the VAST dataset. We extracted these inputs from news stories in the VAST dataset by hand and transformed into a predicate format. Table 2 shows STAB's internal representation of a sample of input events in terms of the knowledge states each event produces.

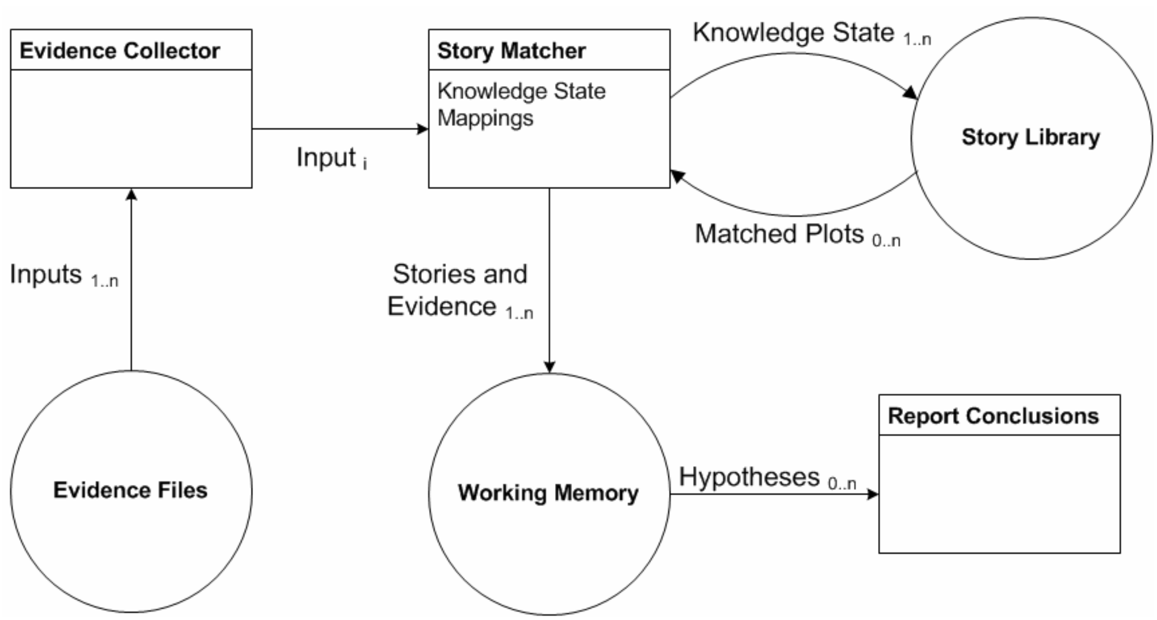

Figure 3: High-Level Architecture of STAB. The processing begins with the evidence collector, which stores the incoming evidence in a chronological order. The story matcher maps the input evidence to knowledge states in the stored plots to retrieve the matching plots from the library. The story matcher then collects the retrieved stories and their supporting evidence in a working memory. Conclusions can be extracted at any point in processing the inputs.

Next, the Story Matcher takes one input event at a time, and uses its resulting knowledge state of the event with the task nodes in the TMKL representations of the story plots stored in the Story Library. The Story Matcher tags the matching tasks and passes the matching stories to a Working Memory. Then, the Story Matcher inspects the next input event in the Evidence File and repeats the above process. If the new input event results in the retrieval of a new plot, then that new plot is similarly stored in the Working Memory.

Sample STAB Inputs
stolen(money $40 Highway-Tire-Store)
cured-disease(Boynton-Labs Philip-Boynton prion-disease)

named-after(lab Philip-Boynton Dean-USC)
was-founded(Boynton-Labs)
have-developed(Boynton-Labs prion-disease)
announced-investigation(USFDA Boynton-Labs)
discontinued-investigation(USFDA Boynton-Labs)
Injected-mouse(Boynton-Labs prion-disease)
Injected-cow(Boynton-Labs prion-disease)
treatment-mouse(Boynton-Labs prion-disease)
treatment-cow(Boynton-Labs prion-disease)

Table 1: This table shows sample inputs for STAB.

If the newly retrieved story is already in the Working Memory, then additional task nodes that match the new input are also tagged but only one story instance is kept. If the input event contradicts the expectations of the knowledge states in the tasks nodes in any story in the Working Memory, then that story is tagged as having been contradicted. In this manner, the expectations of the stories in the Working Memory are either confirmed or contradicted by additional evidence. At any point, the user can query the Working Memory to see the current state of activated hypotheses and the evidence for and against them.

Actions	**Resulting State**
Stolen	Has-object
Broken	Is-broken
Cured-disease	Is-rich-and-famous
Named-after	Expert-involved
Was-founded	Is-open
Have-developed	Exists-new-disease
Announced-investigation	Is-investigating
Discontinued-investigation	Cancel-investigation
Injected-cow	Cow-is-infected
Treatment-cow	Cow-is-cured

Table 2: This table shows the part of STAB's representation of input events that is used by the story matcher to search the story library. The second column shows the knowledge state produced by an event.

4. Analysis of STAB's Sensemaking

In early 2006, PNNL released a synthetic dataset, called the VAST dataset, which captures many of ambiguities and subtleties of the data encountered in real intelligence data. The current dataset contains over a thousand news stories, and a score of tables, maps and photographs. We analyzed this dataset and by screened the stories that indicated an illegal or unethical activity, which left about a hundred new stories out of the more than thousand originally in the dataset. We then analyzed the remaining stories and by hand extracted all the events that pertained to illegal/unethical activities (e.g., Table 1), and handcrafted the internal representations for each of these events in terms of the knowledge states it produces (e.g., Table 2). In addition, we examined the maps, photos and tables that are part of the VAST dataset, and similarly extracted and represented the relevant input information.

Next, we examined the news stories containing illegal/unethical activities, and abstracted story plots from them. We found that seven major story plots appear to cover all the illegal/unethical activities in the VAST dataset. We then represented each of these story plots in the TMKL language. Figure 2 illustrated the Rob a Store plot. Figure 4 illustrates a more complex plot of political conspiracy in which a political figure may get an opponent out of an electoral race by exposing dirt on him or getting him assassinated.

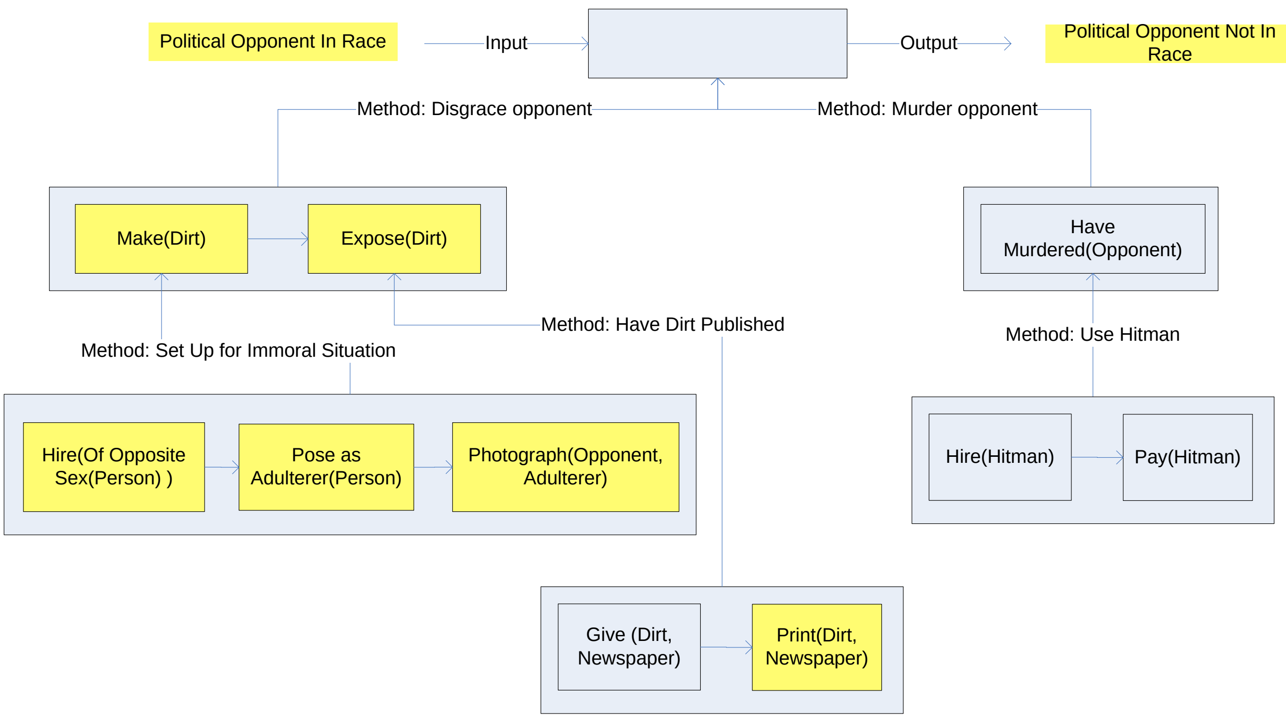

Figure 4: The story plot for a political conspiracy intended to remove an opponent from an electoral race. The task nodes in yellow indicate task nodes that match input events.

As events in the input data stream arrive incrementally, STAB matches the events with the task nodes in the story plots stored in the story library. Figure 5 and 6 illustrates the two story plots, Rob a Store and Commit Vandalism, whose task nodes match the input event Break(Window). The matching task nodes are shown in yellow. Note that Break(Store) provides the intentional context for the even Break(Window). STAB stores both stories (Rob a Store and Commit Vandalism) in its Working Memory as candidate hypotheses.

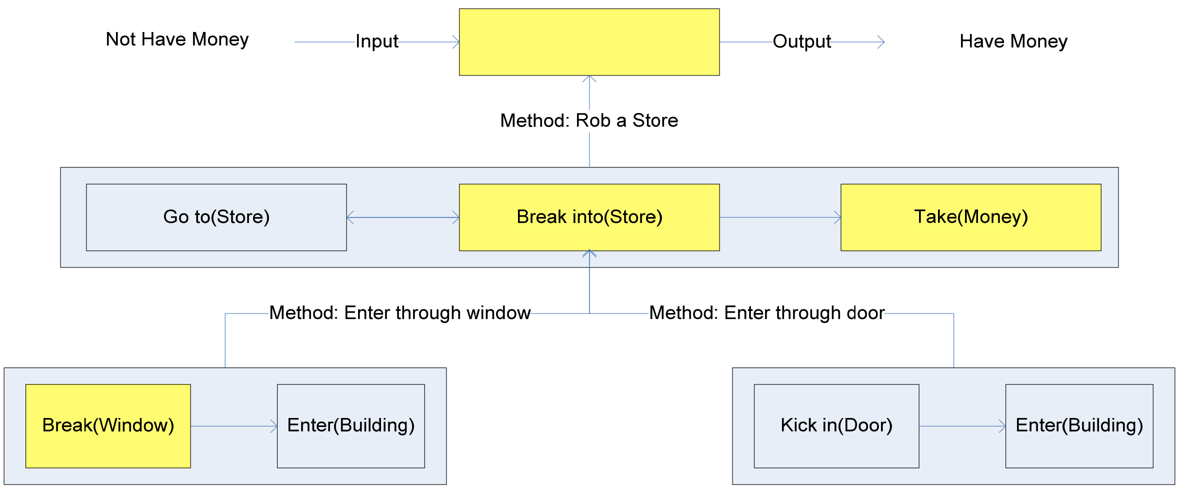

Figure 5: The plot for the Rob a Store story with task nodes that match the input events of Break (Window) and Take (Money) highlighted in yellow. Note that all higher-level task nodes in the plot structure are also highlighted.

The story hypotheses in the Working Memory generate expectations. Thus, Rob a Store generates the expectation of Take(Money) and Commit Vandalism generates the expectation of Stole(Null). As additional data in the form of event Take(Money) arrives as input to STAB, it matches the data with the expectations generated by the candidate hypotheses. This may lead to contradiction of the expectations generated by some hypotheses. For example, in the current scenario, the event Take(Money) results in the contradiction of the expectation of Stole(Null) of the Commit Vandalism hypothesis (shown in red/darker shade). The new data may also lead to confirmation of the expectations of other hypotheses.

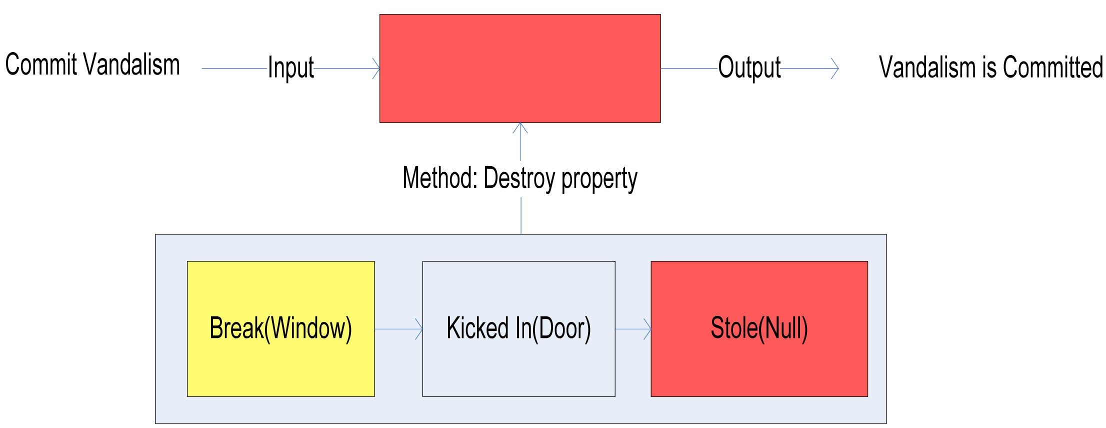

Figure 6: The plot for the Commit Vandalism story. The box with the text Stole(null), shown here in red, is tagged to indicate that this expectation was violated by the data. Note that all higher-level task nodes of this task are also tagged similarly.

5. Discussion

Psychological studies of intelligence analysis [Heuer 1999] indicate the three main errors made by human analysts in generating abductive explanations: (1) Due to limitations of human memory, intelligence analysts may have difficulty keeping track of multiple explanations for a set of data over a long period of time. (2) Analysts may quickly decide on a single explanation for the data set and stick to it as new data arrives. (3) Analysts may look for data that supports the explanation on which they are fixated, and not necessarily the data that may refute their explanation. STAB seeks to address these limitations. Firstly, there no limitations on the size of STAB's Story Library or its Working Memory. On the contrary, STAB offers a non-volatile memory of both generic stories and specific hypotheses. Secondly, for each new additional input event, STAB examines all the stories whose task nodes match the input. Thus, it is not fixated on any particular story at any time. Thirdly, STAB explicitly looks not only for evidence that may confirm the expectations generated by a story hypothesis but also for evidence that may contradict the expectations. In addition, STAB collects the evidence for and against a hypothesis as part of its justification for its conclusions.

However, in its present state of development STAB also has many limitations. Firstly, the input events to STAB are extracted from new stories and represented by hand. In future work, we will build a mechanism for automatic extraction of actions from bew stories. Secondly, STAB's confidence values in its hypotheses are binary, True or False. In current work, we are developing a more robust scheme for computing on the principles of coverage (what proportion of the task nodes in a story plot are matched by the input evidence) and parsimony (what is the level of abstraction of the matched task nodes). Thirdly, we have not yet evaluated STAB's performance against that of a human analyst. Given the high cost of an analyst's time, that evaluation is possible only after STAB becomes much larger and more complex.

6. Conclusions

In Pirroli and Card's [2005] information-processing model of intelligence analysis, the sensemaking loop [Pirolli and Card 2005] involves iterative development of a conceptualization (a *hypothesis*) from stored *schema* that best fits the evidence, and the presentation of the knowledge product that results from this conceptualization. Pirolli and Card, however, do not describe the content and the structure of the schemas in their sensemaking loop, or the process by which specific schemas are conceptualized as hypotheses. STAB provides preliminary, but experimentally verifiable, answers to these questions. According to STAB, the schemas are story plots, where the stories are about the activities of interest to an analyst, e.g., illegal/unethical activities in the VAST domain. The story plots are represented as processes at multiple levels of abstraction, with explicit representation of goals and states. STAB's TMKL language provides one scheme for representation of the story plots. Story plots are retrieved by matching input events with the task nodes in the story plots. The retrieved stories generate expectations about additional data, and the conformation or contradiction of these expectations is a key determinant of the acceptance or rejection of the story hypotheses.

Acknowledgements

This research has benefited from contributions by Andrew Hill, and many discussions with John Stasko, Anita Raja, Neha Sugandh and Tanvi Bhadbhade. This work was sponsored by the National Visualization and Analytics Center (NVAC) under the auspices of the Southeastern Regional Visualization and Analytics Center. NVAC is a U.S. Department of Homeland Security Program led by Pacific Northwest National Laboratory.

References

[Heuer 1999] R. J. Heuer. *Psychology of Intelligence Analysis*. Center for the Study of Intelligence.

[Josephson and Josephson 1994] John R. Josephson & Susan G. Josephson. *Abductive Inference: Computation, Philosophy, Technology*. Cambridge Univerity Press, 1994.

[Krizan 1999] L. Krizan, *Intelligence Essentials for Everyone*, Joint Military Intelligence College, 1999.

[Murdock and Goel 2001] J. W. Murdock & A. K. Goel. Meta-case-Based Reasoning: Using Functional Models to Adapt Case-Based Agents. *ICCBR 2001*: 407-421.

[Pirolli and Card 1999] P. Pirolli and S. Card, Information foraging. *Psychological Review*, 106, 643-675.

[Pirolli and Card 2005] P. Pirolli and S. Card. The Sensemaking Process and Leverage Points for Analyst Technology as Identified Through Cognitive Task Analysis. *Proceedings of 2005 International Conference on Intelligence Analysis*, pp. 2-4, May 2005.

[Sacerdoti 1977] E. Sacerdoti. *A Structure for Plan and Behaviors.* Elsevier-North Holland, Amsterdam.

[Schank and Abelson 1977] R. Schank & R. Abelson. *Scripts, Plans, Goals and Understanding.* Erlbaum, Potomac, MD.

[Thomas and Cook 2005] J. Thomas & K. Cook, Illuminating the Path, 2005.