User-Created and User-Adaptable Technosocial Modeling Methods

Jonathan Pfautz[1], Eric Carlson[1], David Koelle[1] and Emilie Roth[2]
[1] Charles River Analytics Inc.
625 Mt. Auburn Street, Cambridge, MA 02138
[2] Roth Cognitive Engineering
89 Rawson Road, Brookline, MA 02445
{jpfautz | ecarlson | dkoelle}@cra.com, emroth@mindspring.com

Abstract

Over the course of our studies of how intelligence analysis professionals perform their work, we have observed that the perceived utility of technosocial models varies with the degree that analysts could create and/or adapt a model to a particular problem or situation. Analysts failed to trust models (a form of validity) that were created outside of their community of practice. In this paper, we present relevant results from our studies as they pertain to how analysts might create or adapt technosocial models. In particular, we define a number of significant concerns and challenges that must be overcome to enable any sort of user-created or user-adaptable modeling approach. We present a set of tools and techniques that have emerged from our own experiences developing model-based analysis and decision-support systems to illustrate the core challenges and suggest some potential solutions.

Introduction & Background

The utility of a technosocial model that enables predictive analysis is constrained by the degree to which users consider it valid. From a user standpoint, validity might be defined as: generalizability across applications; the amount/accuracy of underlying data used to generate the model; adherence to empirically proven underlying theories (Munson & Hulin 2000); proven ability to replicate human performance in particular domains or situations, as in the AMBR studies (Gluck & Pew 2005); or the degree to which a model represents the real world from the perspective of its intended use (Campbell & Bolton 2005; U.S. Department of Defense 2001).

Over the course of our studies of how intelligence analysis professionals perform their work, we have observed that validity is strongly linked to the user's understanding and acceptance of the composition of the model (e.g., the amount/accuracy of underlying data used to generate the model, the ability to communicate its internal structures and reasoning processes (Young & Harper 2005; Tor et al. 2004)). Enabling end users to verify, modify, and/or create a model's structure increases users' trust in the model, which increases its utility. To fulfill this need, we have developed several techniques that increase model transparency to non-expert users, and that allow these users to adapt and create models.

In addition to increasing trust, easily understood and adapted models double as methods to elicit, capture, and communicate reasoning. Modeling methods can then serve as a direct method of knowledge elicitation, and the resultant models—being easily understood—double as a knowledge representation that can be used for dissemination and as an interactive visualization of reasoning. Easily shared interactive representations of reasoning, in turn, support collaborative modeling.

In this paper, we present the results of our analyses across end-user communities, domains, and applications, allowing us to identify certain commonalities. A key concern of end-users in these domains was the degree to which a particular technosocial model could be either adapted to a current application (e.g., how a model of personality factors could be tied to a mission to influence an adversary leader's decision to use chemical weapons) or created specifically for a given situation (i.e., how a model could be created on the fly to capture key situational knowledge as inputs and outputs, and define relevant internal model constructs—for example, using known intelligence about the frequency of violence in a region, gang participation in that violence, themes in adversary propaganda, and theories of group behavior to create a model of the gangs and forecast their response to increased police action). For a significant number of users, a model would not be considered valid unless it was created by trusted members of the community and adhered to community practices and terminology. This led us to identify a number of challenges and concerns with any approaches to user-created and user-adapted models.

In addition to the results of our analyses, we present the challenges associated with user creation and adaptation of models in the domains we studied and components of in-house modeling tools that we have designed to support varying degrees of end-user model creation/adaptation. These components are not intended to represent a generalized solution, but rather to illustrate our approach, particular challenges, and the range of potential solutions.

Approach

Our approach to analyzing the application of modeling techniques and technologies is grounded in Cognitive Systems Engineering (CSE) practices. CSE represents a principled approach to the design and development of systems based on an analysis of users engaged in work and on an iterative design, implementation, and evaluation cycle (Roth et al. 2002; Woods & Roth 1988; Norman 1986). There are a variety of different methods within CSE (e.g., Cognitive Task Analysis (Schraagen et al. 2000), Cognitive Work Analysis (Vicente 1999), Knowledge Elicitation methods (Cooke 1994)) that share a commitment to understanding the characteristics of the users and the context of work to drive the specification of the entire system design, not just the front-end, user interface components.

In the context of developing systems that employ predictive models, we follow the iterative, user-centric approach represented in Figure 1. While systems employing predictive models can range significantly in their purpose (e.g., decision support, training, acquisition, intelligence analysis), the general process of analyzing end-user needs and iterating on a model and system development remains the same.

Of particular interest within this approach is the need to define the model validity requirements based on an analysis of the end-user. While the validity of a model is defined by the application (i.e., the model must be sufficient for the application), that application is in part determined by how users perform their work. For example, developing a model to embed into a decision-support system that lets a commander forecast adversary tactics necessitates understanding how the commander makes decisions (e.g., Do the decisions need to be made quickly? How much does the commander rely on others' analyses?). This understanding then guides the design of the model and the decision-support system (e.g., How and to what degree is the internal complexity and state of the model presented? How fast does the model return a result?). This understanding should also guide the specification of validity – at what point do imperfections in the model impact the end-user's performance (e.g., If the commander used the predictive model in concert with results from other human-based analyses to confirm/disconfirm the predictions from the model, lower validity may have less of an impact on decision-making performance)? With this approach, we can create a reasonably detailed description of the impacts of technosocial model validity and therefore guide model development and deployment accordingly.

Analysis Results

We have developed, or are developing, software tools for aiding analysis and decision-making with predictive models in a number of domains. In this paper, we present the results of applying a CSE approach across numerous projects, representing thousands of hours with hundreds of domain experts spanning a wide variety of applications, such as military intelligence analysis (Pfautz et al. 2006c), military command and control (Potter et al. 2000; Rosen & Smith 1996; Rimey & Brown 1994), unmanned vehicle control (Pfautz & Roth 2006; Pfautz et al. 2005), sensor network management (Pfautz et al. 2006b), and analysis of weather impacts (Pfautz et al. 2006a). These domains present a disparate set of specific goals but all are concerned with understanding and forecasting behavior. While the general results of these many analyses have been presented elsewhere (Pfautz et al. 2006c; Pfautz et al. 2005), here we focus on how studying these different analyst communities reveals common complexities in the use and validity of predictive models.

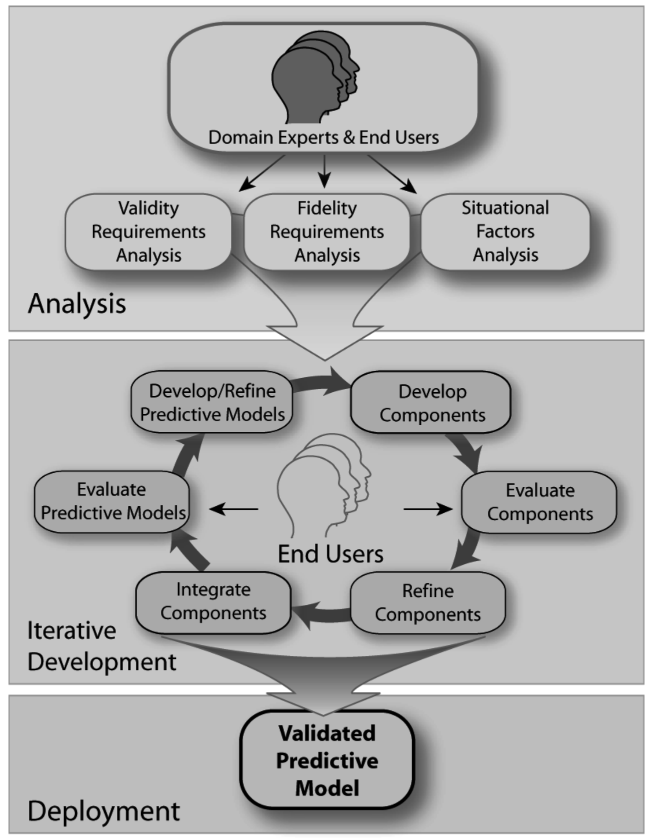

Figure 1: Predictive modeling tools and technology development process

First and foremost, we found that analysts were consistent in their concerns about any model's validity. Few analysts were comfortable with accepting a representation of "what is" or "what if" behavior without some level of understanding of the model's inner workings, including theoretic and computational foundations, data sources used in model construction and application, as well as any and all assumptions in the reasoning design (this defined end-user validity for these domains). Repeatedly, analysts expressed their misgivings about generalized models and the need to express situation-specific qualifications (in the words of one analyst, "All bets are off if it rains on Thursday"). In one domain, analysts vehemently refuted the notion that models could be created by anyone other than analyst community itself – they argued that too much knowledge about the specific situation and about their analysis practices and procedures was required. This perspective, while not as vigorously voiced, was common across analysts—a model was not valid unless it was created by trusted members of the analyst community and adhered to community practices and terminology. This revealed a very specific impact on end-user performance—if the analyst did not trust the model, they would not use it, and therefore there would be no benefit to performance from the model. Trust typically varied with model complexity; the more complex the model and the methodologies used in its creation and application, the less likely that the model would be accepted.

Based on this result, we focused our analysis on two different options: user-created models and user-adaptable models. We identified a number of challenges to be overcome if these options were to be realistic in the domains we studied. We discuss these challenges below.

Challenges of User-Created Modeling

Typically, creating or interpreting models requires knowledge of a range of computational methods, along with an understanding of the various strategies for representing anticipatory analytical thinking. The analysts in the communities we studied were not experts in computational methods. This, therefore, is the key challenge in user-created modeling—users may not have any understanding of the methods and principles used in constructing predictive models. Other related and unrelated challenges include the following needs:

- Computational methods and software tools that enable users to:
    - Identify and select existing entities to model
    - Easily define factors and causal relationships (i.e., externalize their own mental model of predictive factors without sophisticated knowledge of the underlying computation)
    - Provide substantiation for the factors and relationships they express
    - Incorporate particular data sources and provide justification for their selection
    - Create models of varying complexity
    - Evaluate and debug their own models
- Methods for guiding model creation that:
    - Provide a theoretical basis for model structure or components
    - Adhere with community practices and/or doctrine
    - Ensure a consistent and systematic creation process
- Techniques for countering errors or logical inconsistencies in model creation, especially among users unfamiliar with a specific computational method
- Techniques to control for potential user biases in model creation

Challenges of User-Adaptable Modeling

Allowing users to selectively adapt an existing predictive model is subject to the same principle challenge as letting those same users create models: in both cases, the users we studied were not sophisticated modelers. However, while many of the above challenges apply, some are mitigated by the use of "template" models or model components that could be firmly ground in theory and/or captured in a particularly effective computational formalism. Still, some challenging requirements remain with a user-adaptable modeling approach, such as:

- Techniques for defining templates or sub-components of a model that meet end-user criteria for understandability and validity
- Methods for guiding users to the adaptation of templates or components to:
    - Address a particular situation
    - Incorporate their domain-specific knowledge
    - Integrate specific data sources
    - Adhere with community practices
    - Build and expand constantly evolving domain-specific terms and definitions
    - Collaborate with others on adaptation/expansion of a model

Tools and Techniques

Given a number of efforts to develop reasoning and decision support systems for intelligence analysts, we developed specific tools and techniques that represent some potential solutions to the above challenges. Perhaps the most critical challenge was to identify or develop computational methods and software tools that enable user-created modeling. After considering rule-based methods, fuzzy logic, and other formalisms, we chose Bayesian belief networks (BBNs) as particularly promising (Pearl 2001). BBNs support the creation of graphical models—that is, they allow users to create variables, link them together, and specify probabilistic relationships in a graphical environment. They represent a significant degree of computational sophistication (allowing for deductive and abductive reasoning, reasoning under uncertainty, etc.). However, as noted in prior work (Pfautz et al. 2007), BBNs are still not completely amenable to user-created modeling because of a number of specific issues (e.g., exponential growth of entries needed in conditional probability tables as the number of parents and parent states increase). To address these issues, we developed a new computational formalism, Causal Influence Models

(CIMs) which represent an improvement to existing Influence Diagram (Howard & Matheson, 984) and Influence Network (Rosen & Smith 1996) techniques. An example CIM is shown in Figure 2.

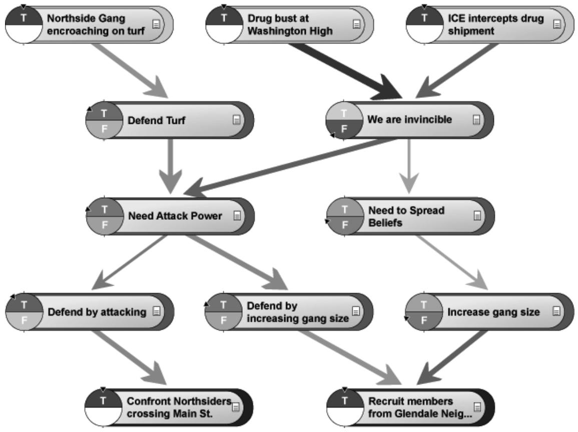

Figure 2: Example CIM representing a hypothetical group's attitudes, beliefs, and goal scripts

We implemented our CIM approach within a pre-release version of our BNet.Builder™ software. We deliberately considered CIMs in a static case—not as models within a high fidelity modeling and simulation environment, but rather as specific tools to aid an analyst in: (1) reasoning about the current state of a system (i.e., "what is"), and (2) speculating about changing one or more factors and observing its impact on the system's behavior (i.e., "what if"). Over the course of multiple projects, we worked with domain experts (on the order of 40-50 individuals) to see if they could externalize their knowledge with this formalism, working specifically in the domain of individual and group behavior. We found that they could easily define factors and causal relationships (i.e., create some limited behavior representations) with minimal training (on the order of hours), but that the open-ended nature of the modeling meant that they were at a loss for a "starting point."

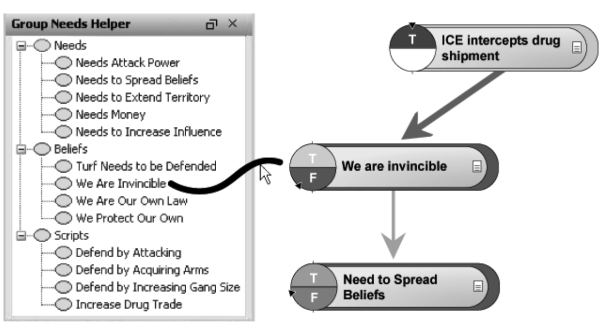

Figure 3: Providing theory- or practice- based guidance for intermediate structure development.

We approached this problem by providing a very simple structure to help guide users to identifying inputs and outputs to the intermediate model structure. Color-coding, with supporting network layout heuristics, allowed the users to more cleanly define the "stimuli" and "actions" or "behaviors" of the model. While this helped to a degree, it still left the intermediate structure of the model relatively open. Therefore, our next step was to introduce the concept of "helpers," tools that would allow for the specification of intermediate model structure from existing or prior theories and practice. For example, we might specify (via an easily swapped ontology) a set of personality factors for an individual or a set of cultural influences for a group. These factors can then be easily dragged-and-dropped into the CIM, as shown in Figure 3.

We augmented this approach by providing the ability to create additional structure after a factor was initially identified (e.g., this structure might represent a more complicated underlying model, or may simply help identify additional correlated factors that the analyst should consider). This allows us to provide a basis in theoretical Behavioral Science (e.g., the 5-factor model) or, more operationally, in experts' past experiences (e.g., recent patterns of gang behavior in the area of operation), as needed in the domain to achieve application validity. It also can enable a systematic and/or consistent application of theory or practice, depending on the restrictions that are placed on how factors can be created via the drag-and-drop interface.

Another important step in supporting user-created modeling was identifying the system being modeled, along with its relationship with external factors and entities, which we support through a typical network view of entities and their relationships (based on our in-house social network analysis tool, CONNECT™), as shown in Figure 4.

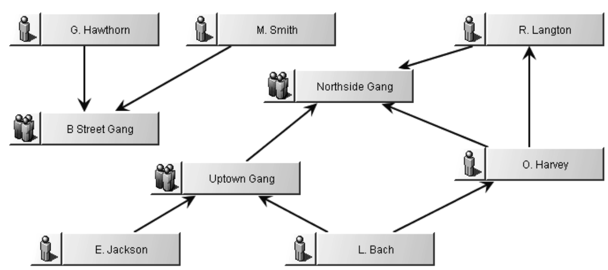

Figure 4: Network via of entities of interest

Via this interface, analysts could specify the system or reasoning process for which they wanted to create a model. This allows the analyst to be very situation-specific in how they construct and apply their models.

Another key challenge to address was the user's need to ensure that the model they created was adequately based on substantiated information regarding a particular individual or group. To support this need, we developed an interface to a user-defined file structure containing relevant information. Within this interface, the analyst can choose a document and/or select a specific fact or conclusion from the document, and drag-and-drop that information as an input or output of the model (e.g., this model is seeing an increase in bombings in this area, this model predicts an upcoming offensive in the general area). This tool is shown in Figure 5.

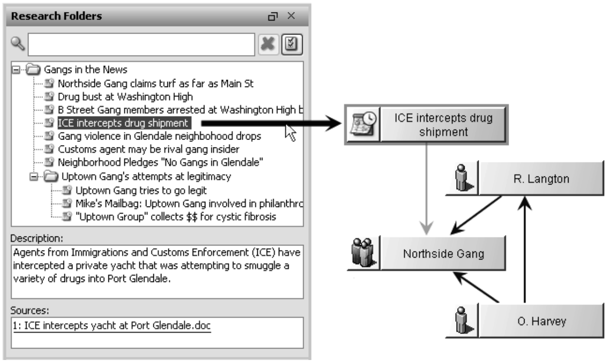

Figure 5: Allowing analysts to specify model inputs and outputs from existing research or data sources

This tool also allows the variables and links in the CIM to be auto-annotated with data source information and/or the comments of the model creator, as shown in Figure 6.

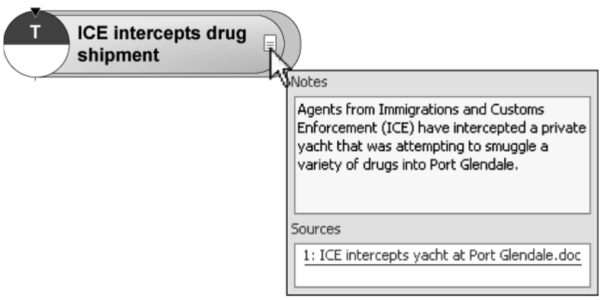

Figure 6: Annotations on a node showing auto-populated sources and user comments

The ability to annotate the model in this fashion provides an additional capability important to the user communities we studied: validation through collaboration. While analysts justify their reasoning through their real-world experience, they also actively vet their reasoning with others to help ameliorate individual biases, identify additional/missing pieces of a model, and ensure sources and justifications are sufficient. This community-of-practice-based validation is done through a number of mechanisms, including collaborative model construction, model review by superiors, and model review by experts. This last step is of particular interest because it provides the opportunity for experts to assess how a particular theory is applied in a particular situation. In the domains we studied, we found many processes for ensuring analytic rigor (Zelik et al. 2007), and therefore worked to construct tools to support these existing practices by which models of human behavior could be validated.

The simplest form of collaboration for validation involved supporting a small group of model builders working collaboratively. The CIM's interface not only allows for rapid specification and connection of factors, but also permits users in actively testing and refining their models. This allows for a group to immediately see the model that is created and then walk through test cases based on their own experiences to ensure model validity. We regularly incorporate collaborative communication tools (e.g., chat, email) into the design of our systems to help with situations where model constructors and evaluators are not co-located. We also have developed a version of our CIM that allows for analysts to simultaneously view and edit (via token-based collaboration) models.

We found that the ease with which users could create linkages while building models with the CIM could lead to logistical errors. Therefore, we introduced a tool that translates paths through the CIM into prose, aiding in validation and verification of the model either by the creator themselves or by a supervisor or domain expert. This tool is shown in Figure 7.

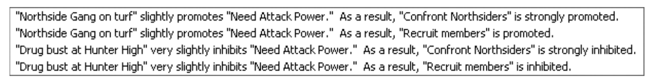

Figure 7: A tool for user-review of logical pathways in a CIM-based model

This verification and validation tool, in addition to relatively commonplace collaboration tools, supports a community of practice approach to model creation. This, in turn, supports the concept of user-adaptable modeling, where models developed and/or validated elsewhere in the community of practice can be manipulated by analysts to suit a particular situation or mission. The details of a model can be provided via the annotation tools, allowing model creators to share their reasoning about particular structures with model adaptors and therefore reducing the misapplication of models. Existing collaboration tools can be similarly used in a user-adaptive modeling approach to allow Behavioral Science experts or more experienced analysts to assess and refine the application of a model to a particular situation (i.e., to provide an independent assessment of application validity).

In both user-defined and user-adapted models, experimentation is a common method used by analysts to understand and trust the implications of a set of factors and relationships. During model creation, analysts will often experiment with permutations of evidence to confirm adherence to some hypothesis as method establishing correctness and completeness. While the verification and validation tool supports this behavior by clearly delineating which factors are related, active experimentation remains an important technique. To support this, the model development environment updates in real-time, allowing analysts to actively observe model behavior in reaction to changes in evidence. To further support this practice, we introduced a tool that allows users to monitor changes to the beliefs of factors in the network. Using this tool, analysts can set the current status of a model as a baseline. Deviations from this baseline are displayed as evidence on one or more nodes is altered. This belief monitor tool is shown in Figure 8.

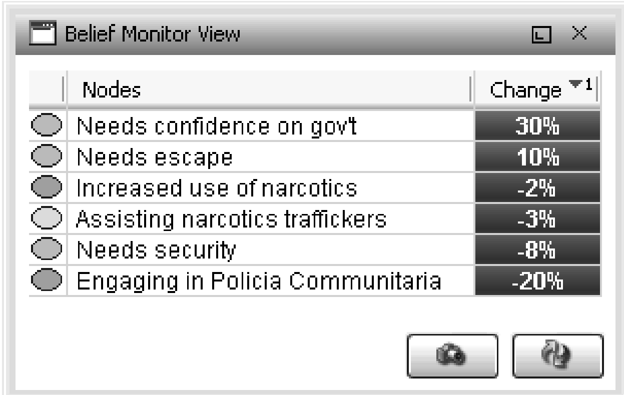

Figure 8: A tool to view changes to the beliefs of factors in the model from a user-specified baseline

Conclusions

The concept of user-created or user-adapted models should necessarily be a source of concern for the predictive modeling community. As described above, there are a number of considerable challenges in bringing computationally or theoretically sophisticated modeling capabilities to operational user communities with little or no expertise in interacting directly with the underlying composition of a model. We have presented a set of tools and techniques that attempt to address these challenges, but recognize that a number of these challenges are a function of the community using the tools and their strategies for managing model validity. This end-user focus, derived from Cognitive System Engineering (or general requirements analysis) helps identify the bounds of application validity for a particular domain, and can provide scope for the construction of analysis or decision-support tools. In our work, we found communities with existing practices for describing, qualifying, and validating models, which led us to develop tools to support these practices, rather than to the development of complex models. Given the support of these tools, we have found that analysts can successfully create and adapt predictive models, and that this can increase understanding and acceptance—and, therefore, the utility—of these models to analysts.

We anticipate future work to extend the approaches described above, particularly towards more complex modeling techniques (e.g., agent-based simulation). We also plan to explore the degree to which other computational methods are amenable to user expression of a model (e.g., leveraging research on how experts express their judgments (O'Hagan et al. 2006)). Finally, we plan to study the interaction between model complexity and perceived end-user validity in more detail, investigating trust and utility trade-offs as a function of risk/consequence management (e.g., low tolerance for less perceived validity in situations with significant adverse consequences).

Acknowledgements

The authors would like to express their deepest gratitude to the many dedicated analysts who we interviewed and observed and who must remain anonymous. The concepts and solutions presented here were funded across multiple efforts, including work performed, in some part, under Government Contract Nos. FA8650-04-C-6403, W15P7T-06-C-C001, and FA8650-06-C-6731 by a number of scientists and engineers at Charles River Analytics, including Geoffrey Catto, Joseph Campolongo, Amanda Seybold, and Sofya Tenenbaum.

References

Campbell, G. & Bolton, A. (2005). HBR Validation: Integrating Lessons Learned From Multiple Academic Disciplines, Applied Communities, and the AMBR Project. In R. Pew & K. Gluck (Eds.), Modeling Human Behavior With Integrated Cognitive Architectures (pp. 365-395). Mahweh, NJ: Lawrence Earlbaum Associates.

Cooke, N. M. (1994). Varieties of Knowledge Elicitation Techniques. International Journal of Human-Computer Studies, 41(6), 801-849.

Gluck, K. & Pew, R. (2005). Modeling Human Behavior With Integrated Cognitive Architectures. Mahweh, NJ: Lawrence Erlbaum.

Howard, R. & Matheson, J. (1984). Influence Diagrams. In R. Howard & J. Matheson (Eds.), Readings on the Principles and Applications For Decision Analysis (pp. 719-762). Menlo Park, CA: Strategic Decisions Group.

Munson, L. & Hulin, C. (2000). Examining the Fit Between Empirical Data and Theoretical Simulations. In D. Ilgen & C. Hulin (Eds.), Computational Modeling of Behavior in Organizations (pp. 69-83). Washington, D.C.: Am. Psychological Association.

Norman, D. (1986). Cognitive Engineering. In D. Norman & S. Draper (Eds.), User-Centered System Design. Hillsdale, NJ: Lawrence Erlbaum Associates.

O'Hagan, A., Buck, C., Daneshkhah, A., Eiser, R., Garthwaite, P., et al. (2006). Uncertain Judgements: Eliciting Experts' Probabilities. New York: Wiley & Sons.

Pearl, J. (2001). Causality: Models, Reasoning, and Inference. Cambridge University Press.

Pfautz, J., Cox, Z., Koelle, D., Catto, G., Campolongo, J., & Roth, E. (2007). User-Centered Methods for Rapid Creation and Validation of Bayesian Networks. In Proceedings of 5th Bayesian Applications Workshop at Uncertainty in Artificial Intelligence (UAI '07). Vancouver, British Columbia.

Pfautz, J., Fouse, A., von Wiegand, T., Roth, E., & Fichtl, T. (2006a). An Approach for Multimodal Display Development for UAV Systems. In Proceedings of 3rd Annual Workshop on Human Factors of Unmanned Aerial Vehicles. Mesa, Arizona.

Pfautz, J. & Roth, E. (2006). Using Cognitive Engineering for System Design and Evaluation: A Visualization Aid for Stability and Support Operations. International Journal of Industrial Engineering, 36(5), 389-407.

Pfautz, J., Roth, E., Bisantz, A., Fouse, A., Madden, S., & Fichtl, T. (2005). The Impact of Meta-Information on Decision-Making in Intelligence Operations. In Proceedings of Human Factors and Ergonomics Society Annual Meeting. Orlando, FL.

Pfautz, J., Roth, E., Bisantz, A., Llinas, J., & Fouse, A. (2006b). The Role of Meta-Information in C2 Decision-Support Systems. In Proceedings of In Proceedings of Command and Control Research and Technology Symposium. San Diego, CA.

Pfautz, J., Roth, E., Powell, G., Fichtl, T., Guarino, S., & Carlson, E. (2006c). Cognitive Complexities Impacting Army Intelligence Analysis. In Proceedings of Human Factors and Ergonomics Society 50th Annual Meeting. San Francisco, California.

Potter, S. S., Roth, E. M., Woods, D. D., & Elm, W. C. (2000). Bootstrapping Multiple Converging Cognitive Task Analysis Techniques for System Design. In Chipman, Shalin, & Schraagen (Eds.), Cognitive Task Analysis (pp. 317-340). New Jersey: Lawrence Erlbaum.

Rimey, R. & Brown, C. (1994). Control of Selective Perception Using Bayes Nets and Decision Theory. International Journal of Computer Vision, 12(2-3), 173-207.

Rosen, J. & Smith, W. (1996). Influence Net Modeling With Causal Strengths: An Evolutionary Approach. In Proceedings of Command and Control Research and Technology Symposium.

Roth, E. M., Patterson, E. S., & Mumaw, R. J. (2002). Cognitive Engineering: Issues in User-Centered System Design. In J. J. Marciniak (Ed.), Encyclopedia of Software Engineering (Second ed.), (pp. 163-179). New York: Wiley Interscience, John Wiley and Sons.

Schraagen, J. M. C., Chipman, S. F., & Shalin, V. L. (2000). Introduction to Cognitive Task Analysis. In J. M. C. Schraagen, S. F. Chipman, & V. L. Shalin (Eds.), Cognitive Task Analysis (pp. 3-24). Mahwah, NJ: Lawrence Erlbaum Associates.

Tor, K., Ritter, F., & Haynes, S. (2004). CaDaDis: A Tool for Displaying the Behavior of Cognitive Models and Agents. In Proceedings of Behavior Representation in Modeling and Simulation.

U.S. Department of Defense (2001). VV&A Recommended Practices Guide Glossary. Washington, D.C.: Defense Modeling and Simulation Office.

Vicente, K. J. (1999). Cognitive Work Analysis: Towards Safe, Productive, and Healthy Computer-Based Work. Mahwah, NJ: Lawrence Erlbaum Associates.

Woods, D. D. & Roth, E. M. (1988). Cognitive Systems Engineering. In M. Helander (Ed.), Handbook of Human-Computer Interaction. New York: North-Holland.

Young, A. & Harper, K. A. (2005). TRACE: An Ontology-Based Approach to Generic Traceability Tools for Human Behavior Model. In Proceedings of 2005 Behavior Representation in Modeling and Simulation. Universal City, CA: DMSO.

Zelik, D., Patterson, E., & Woods, D. (2007). Judging Sufficiency: How Professional Intelligence Analysts Assess Analytical Rigor. In Proceedings of Human Factors and Ergonomics Society 51st Annual Meeting.

Author Biographies

JONATHAN PFAUTZ is a Principal Scientist and Vice President of the Cognitive Systems Division at Charles River Analytics Inc. in Cambridge, MA. His major research interests include social network analysis, individual differences, cultural anthropology, intelligence analysis, human-computer interfaces, and cognitive task analysis-based system development for a variety of military and non-military domains. He received his Ph.D. from Cambridge University.

ERIC CARLSON is a Scientist at Charles River Analytics Inc. His research interests include designing human-in-the-loop automation, information and knowledge representation, human behavior modeling, social network analysis, and multi-agent systems. Mr. Carlson received his B.S. in Cognitive Science from University of California, San Diego.

DAVID KOELLE is a Senior Software Engineer at Charles River Analytics Inc. He is the lead engineer on a number of efforts to create intelligence analysis tools for describing and forecasting socio-cultural influences on human behavior. Mr. Koelle received his B.S. in Computer Science from Worcester Polytechnic Institute.

EMILIE ROTH is a Cognitive Psychologist, Human Factors Engineer, and Founder of Roth Cognitive Engineering in Boston, MA. She is an expert in cognitive task analysis and naturalistic decision making in dynamic high-risk environments and is considered a pioneer in the field of Cognitive Systems Engineering. She received her Ph.D. from University of Illinois at Urbana-Champaign.