A General Instance-Based Model of Sensemaking in a Functional Architecture

Robert Thomson, Christian Lebiere and Stefano Bennati
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15218
412-268-6284
thomsonr@andrew.cmu.edu, cl@andrew.cmu.edu, sbennati@andrew.cmu.edu

Keywords:
Sensemaking, Decision-Making, Instance-Based Learning, Functional Modeling, Cognitive Architectures

ABSTRACT: *This paper describes a general instance-based learning model of sensemaking in the context of geospatial intelligence tasks. Building upon a model previously described in Lebiere, Pirolli, Thomson et al. (2013), our model captures human performance across two tasks involving generating and updating likelihoods based on simulated geospatial intelligence. The model predicted human performance in such cognitive functions as generating and updating likelihoods based on incoming information, and in hypothesis/strategy selection and updating based on likelihoods taken in the context of experiences learned from prior exemplar. We then describe an initial attempt at a general instance-based model of decision-making capable of performing any task describable as a directed graph.*

Introduction

In this paper, we describe a computational cognitive model, developed in the ACT-R architecture, of several core hypothesis-generation and updating processes involving *sensemaking* in a simulated intelligence analysis task (called TACTICS). Sensemaking, as in *to make sense of*, implies an active process of constructing a meaningful and functional representation of some aspects of the world (a *frame*) with the goal of completing some actionable outcome (i.e., making a decision).

The problem of biases in intelligence analysis is a pressing concern, with failures in reasoning leading to negative diplomatic repercussions and inefficient operation of limited resources. One issue with treating biased behavior as a *failure* in reasoning is that, under naturalistic conditions, many heuristics that are the source of biased behavior are in fact adaptive and effective. As such, we argue that we study the misapplication of rational intuitive heuristics to what is inherently an artificial domain (i.e., intelligence analysis).

In our tasks, rational Bayesian optima are defined over probability judgments, with cognitive biases defined as being deviations from these optima. Heuristics are then imputed from regularities in these deviations. As will be discussed, participants tended to a adopt an *averaging* heuristic when presented with two probabilities; that is, they treated the probabilities like independent sources of evidence and estimated the outcome to be as likely as the arithmetic mean between the two sources.

The structure of this paper will be as follows. The remainder of the introduction will provide overviews of sensemaking and ACT-R, and a description of prior research using an ACT-R instance-based learning model. This unified model captured human behavior across a series of six sensemaking tasks. We will then describe the current TACTICS tasks, the current ACT-R model, and provide quantitative fits to the human data. Finally, we will describe a general-purpose ACT-R decision-making model theoretically capable of performing any sensemaking task describable as a directed graph.

The Data/Frame Theory of Sensemaking

According to Klein, Moon, & Hoffman (2006), a frame is a limited capacity mental structure (3-4 datum) used to both index existing data and to guide foraging for additional data (e.g., the elaborating cycle; see Figure 1). A frame reflects the compiled prior experiences of an individual. Since we can only attend to a small portion of the available information in our environment, we are constantly in a process of gathering data to support our current frame, while at the same time having our current frame determining which data are noticed.

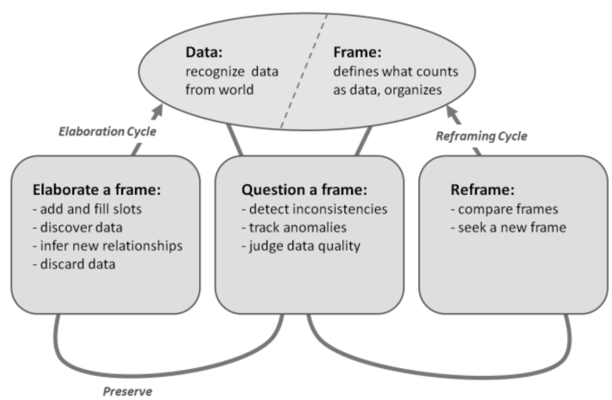

*Figure 1.*The Data/Frame theory assumes that meaningful mental representations called *frames* define what is considered *data* and how this data is structured for mental processing. Image reproduced from Klein et al., 2006.

Whereas frames define and shape existing evidence, incoming data can also evoke changes to one's currently held frame. Hypotheses drive top-down processes such as guiding attention to relevant information through the application and interpretation of frames. For instance, the frame for a house fire is different whether you are the homeowner, the firefighter, or the arson investigator (Klein et al., 2006).

Sensemaking can involve the elaboration of a frame (e.g., filling in details), questioning a frame (e.g., due to the detection of anomalies), or reframing (e.g., rejecting a frame and replacing it with another). The Data/Frame theory proposes that backward-looking processes are involved in forming mental models that explain past events, and forward-looking mental simulations are involved in predicting how future events will unfold.

The ACT-R 6 Architecture

ACT-R 6 is a computational implementation of a unified theory of cognition (Anderson, Bothell, Byrne, et al., 2004). It accounts for information processing in the mind via task-invariant mechanisms constrained by the biological limitations of the brain (see Anderson, 2007 for an overview). While sensemaking theory abstracts away from brain processes, it makes commitments to the control and flow of information that are commensurable with ACT-R's functional perspective. For example, the elaboration and reframing loops in sensemaking can be instantiated in the production rules controlling the flow of control and information in ACT-R. Furthermore, ACT-R is committed to localization of neural architecture, allowing for functional models to guide the development of neurally-inspired models (e.g., Lebiere, Pirolli, Thomson, et al., 2013).

The ACT-R architecture is organized as a set of modules, each devoted to processing a particular kind of information, which are integrated and coordinated through a centralized production system module. Each module is assumed to access and deposit information into buffers associated with the module, and the central production system can only respond to the contents of the buffers, not the internal encapsulated processing of the modules. For instance, the goal module stores and retrieves information that represents the internal intention and problem solving state of the system and provides local coherence to behavior.

The declarative memory and production system modules, respectively, store and retrieve information that corresponds to *declarative knowledge* and *procedural knowledge*. Declarative knowledge is the kind of knowledge that a person can attend to, reflect upon, and usually articulate in some way. Procedural knowledge consists of the skills we display in our behavior, generally without conscious awareness. Declarative knowledge in ACT-R is represented formally in terms of chunks. The information in the declarative memory module corresponds to personal episodic and semantic knowledge that promotes long-term coherence in behavior. In this sense a chunk is like a data frame, integrating information available in a common context at a particular point in time in a single representational structure.

Chunks are retrieved from long-term declarative memory by an activation process. When a retrieval request is made to declarative memory (DM), the most active matching chunk is returned, where activation is computed as the sum of base-level activation, spreading activation, mismatch penalty and stochastic noise.

Each chunk has a base-level activation that reflects its recency and frequency of occurrence. Activation spreads from the current focus of attention, including goals, through associations among chunks in declarative memory. These associations are built up from experience, and they reflect how chunks co-occur in cognitive processing. The spread of activation from one cognitive structure to another is determined by weighting values on the associations among chunks.

Chunks are also compared to the desired retrieval pattern using a partial matching mechanism that subtracts from the activation of a chunk its degree of mismatch to the desired pattern, additively for each component of the pattern and corresponding chunk value. Finally, noise is added to chunk activations to make retrieval probabilistic, governed by a Boltzmann distribution.

While the most active chunk is usually retrieved, a blending process (Lebiere, 1999) can also be applied that returns a derived output reflecting the similarity between the values of the content of all chunks, weighted by their retrieval probabilities reflecting their activations and partial-matching scores. This blending process will be used extensively in the model since it provides a tractable way to learn to perform decisions in continuous domains such as probability space.

The flow of information is controlled in ACT-R by a production system, which operates on the contents of the buffers. Each production consists of if-then condition-action pairs. Conditions are typically criteria for buffer matches, while the actions are typically changes to the contents of buffers that might trigger operations in the associated modules. The production with the highest utility is selected to fire from among the eligible productions. Please see Anderson and Lebiere (1998) and Anderson et al. (2004) for a more complete account of the mechanisms implemented in the ACT-R architecture.

ACT-R and Instance-Based Learning

Instance-based learning theory (IBL; Gonzalez, Lerch, & Lebiere, 2003; Taatgen, Lebiere, & Anderson, 2006) is the claim that implicit expertise is gained through the accumulation and recognition of experienced events or instances. IBL was formulated within the principles and mechanisms of cognition in ACT-R, and makes use of the dynamics of chunk retrieval and blended retrievals.

The main claim of IBL is that implicit knowledge is generated through the creation of instances. These instances are represented in chunks with slots containing the conditions (e.g., a set of contextual cues), the decision made (e.g., an action), and the outcome of the decision (e.g., the utility of the decision). Before there is sufficient task-relevant knowledge, decision-makers implicitly evaluate alternatives using heuristics (e.g., random choice, minimize loss, maximize gain). Once a sufficient number of instances are learned, decision-makers retrieve and generalize from these instances to evaluate alternatives, make a decision, and execute the task.

The process of feedback involves updating the outcome slot of the chunk according to the post-hoc generated utility of the decision. Thus, when decision-makers are confronted with similar situations while performing a task, they gradually abandon general heuristics in favor of improved instance-based decision-making processes (Gonzalez & Lebiere, 2005).

IBL offers constraints on explanation by grounding implicit learning within the mechanisms of a cognitive architecture. For instance, the dynamics of an instance's sub-symbolic activations (e.g., frequency and recency in the base-level activation equation) provide a scientifically-justified mechanism for determining which instances are likely to be retrieved for a given situation, and also can explain *why* they were retrieved and what factors came into play. This provides a much more rigorous explanation of intuitive decision-making than case-studies and introspection of experts.

Models related to decision-making and problem-solving models in ACT-R over the past 10 years have seen increasing use of IBL (whether explicitly referred-to as such or otherwise; e.g., Kennedy & Patterson, 2012) to learn intuitive knowledge structures. This is unsurprising given that ACT-R's declarative memory module and chunk structure is an excellent match for the storage and retrieval of instances, which effectively guides people to some form of IBL. In other words, the design and constraints of the architecture lead people to adopt an IBL-like approach by using the architecture in the most direct and intuitive way.

IBL methodology has been used in a number of research applications including the AFRL 711 HPW/ RHA's model of Predator operators. It can also be used to represent individual differences in experience and capacity by providing and parameterizing content from a single individual (e.g., Sanner et al., 2000; Wallach & Lebiere, 2003).

Rather than provide an overview of many examples, we would like to focus on an in-depth analysis of a single ACT-R model of sensemaking that uses IBL to perform multiple complex geospatial intelligence tasks and provides both an explanation of biases and a close fit to human data (see Lebiere et al., 2013, for a more complete description of the tasks and quantitative model fits).

The ICArUS Challenge Tasks Model

The ICArUS Challenge Tasks were a series of six complex simulated geospatial intelligence tasks, and was composed of three sequential components. The first was focused on learning statistical patterns of events and then generating probability distributions of category membership based on the spatial location and frequency of these events (e.g. how likely does a given event belong to each of the categories). The second required the application of probabilistic decision rules in order to generate and revise probability distributions of category membership (e.g., if a given feature is present at an event, then that event is twice as likely to belong to category A). The third involves making decisions about the allocation of resources based on the judged probabilities of the causes of perceived events, and was effectively a metacognitive measure of confidence in one's judgment. For more detail, please see Lebiere et al., (2013) and Thomson et al., (2012).

The model perceived events and stored them in declarative memory as instances. In all tasks, probability adjustment and resource allocation were performed using a common instance-based learning mechanism, with experience from earlier tasks accumulated in memory for use in later tasks. To leverage the IBL approach for probability adjustment, the model's memory was populated with a range of facts consisting of triplets: an initial probability, an adjustment factor, and the resulting probability. These triplets correspond roughly to the notion of a decision frame.

The adjustment factor was set by the explicit rules of the task (e.g., an event in a category boundary is twice as likely to belong to that category). The model was then seeded with a set of chunks that correspond to a range of initial probabilities and an adjustment factor together with the posterior probability that would result from multiplying the initial probability by the adjustment factor, then normalizing. When the model is asked to estimate the resulting probability for a given prior and multiplying factor, it simply performed a blended retrieval specifying prior and factor, and outputted the posterior probability that represented the blended consensus of the seeded chunks.

Resource allocation was also performed in all tasks using the same instance-based approach, with results from earlier tasks fundamentally affecting choices in later tasks. Representation of a trial instance consisted of three parts: a decision context (in this case, the probability of the leading category), the decision itself (i.e., the resource allocation to the leading category), and the outcome of the decision (i.e., the payoff resulting from the match of that allocation to the ground truth of the identity of the responsible category). The remaining resources were divided amongst the remaining categories in proportion to their assigned probabilities. This unified mechanism has no explicit strategy, but instead learns to allocate resources according to the outcome of prior decisions.

The integrated ACT-R model performed all 6 tasks using the same knowledge constructs (production rules and chunks, other than those it learns as part of executing the task) and parameters. The model was run the same number of times as participants in the dataset (45) with the average model response compared to the average human performance. The natural variability in the model (stochastic elements influencing instance-based learning) approximates some of the individual differences of the human participants. The average fit to human data across tasks was excellent, $r^2 = .756$; with the model predicting trial by trial variability in almost all trials. In addition, when comparing the model against humans in terms of biased behavior (using a *negentropy* measure), the model predicted not only the existence, but also the magnitude of four biases (confirmation, anchoring and adjustment, probability matching, and base-rate neglect), $r^2 = .645$. The model was then compared against a second human dataset using a different exam and performed similarly, justifying the overall fit to human performance.

The TACTICS Tasks

The TACTICS tasks are designed to study the role of cognitive biases in sensemaking in the context of intelligence analysis. They are the successor to the ICArUS challenge tasks, extending intelligence analysis to the realm of adversarial multi-choice paradigms. The general flow of a trial is as follows: gather intelligence from the display and make a probability judgment of the likelihood that a group (red) will attack, then gather more intelligence and revise your judgment. At the end of the trial, make a final probability (of attack) and determine (based on likelihood of victory and payoff) whether to meet the attack or to divert the attack away. An example of the display is presented in Figure 2.

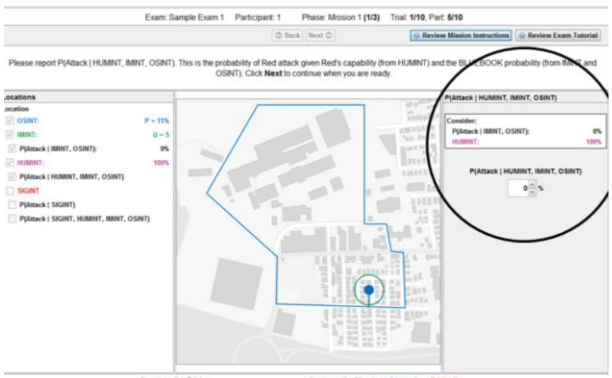

*Figure 2.*An example of the mission display in the TACTICS tasks. The left panel is a legend, and the right shows probabilities and is where responses are entered.

An overview of the intelligence and steps used in the tasks is provided in Table 1.

Table 1. Overview of Intelligence and steps in Task 1.

Symbol	**Meaning**
P	probability that Blue will defeat Red (i.e., Blue's Vulnerability) provided by OSINT
U	Utility/payoff at stake in a showdown (i.e., the Opportunity), provided by IMINT
P_c	probability that Red has the capability to attack, P_c, provided by HUMINT
P_p	probability that Red has the propensity to attack, P_p, given the capability to attack
$P_{p,c}$	probability that Red has the propensity and capability to attack, $P_{p,c}$
P_t	probability of Red attack as signaled by Red Signals Intelligence (SIGINT), P_t
$P_{t,p,c}$	probability of Red attack, per activity, propensity, and capability
D	divert, an action by Blue

In task 1, participants are initially provided with OSINT (P) and IMINT (U), and are required to generate a propensity (P_p) of attack by looking up red's initial probability of attack in a BLUEBOOK (see Figure 3). Then, red's capability of attacking (P_c) is revealed (0-100%) and participants are instructed to generate the joint probability $P_{p,c}$ based on propensity and capability. Participants then are shown SIGINT (signal intelligence) which shows the presence or absence of chatter at the possible attack location. Based on the grid shown in Figure 3 (right), participants enter the likelihood of P_t. Participants are then instructed to calculate the joint probability of P_t and $P_{p,c}$, which is the final probability of red attack, $P_{t,p,c}$. Finally, based on $P_{t,p,c}$, the participant must choose whether to divert the attack or meet the attack for a showdown, with the payoffs based on U.

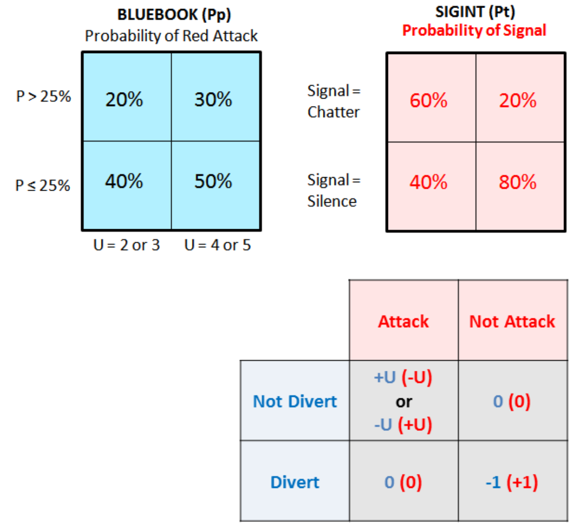

Figure 3. Sample Bluebook (P_p), SIGINT (P_t), and payoff grid provided to participants. In the payoff matrix, values in red pertain to red decisions and gains or losses.

In task 2, there is an additional step. Participants are instructed that there are two potential red strategies, passive and aggressive, each with their own Bluebook. Starting in trial 2, participants must choose which strategy they believe red is using. They are instructed that red's strategy does not shift during the task. Both tasks run for 10 trials, and 30 participants completed the tasks.

The TACTICS Model

The current model builds on the ICArUS Challenge Tasks integrated model using a common instance-based learning approach in probability adjustment and resource allocation. The components dealing with the unique aspects of TACTICS, specifically the adversarial multi-choice paradigm, build on a series of models of similar tasks that have been built in ACT-R and validated against human data. Those models include two distinct paradigms: forced-choice tasks with probabilistic payoffs, and adversarial game-playing with discrete options. A recent experimental study that manipulated information conditions on a spectrum across the two paradigms indicated the potential for unifying these two paradigms.

The first paradigm, forced-choice tasks with probability payoffs, has been modeled and applied to a number of data sets, most prominently by winning the Technion Prediction Tournament. This competition required models to predict subject choices for a range of payoff distributions for which data had been withheld (Erev et al., 2010). The model worked by representing the association between each option and its numerical payoff in DM. The model then generates its expectation for each option through a blended retrieval, the same mechanism used for probability revision and resource allocation in the ICArUS Challenge Tasks model. The option with the highest expected payoff is chosen.

The second paradigm, adversarial game playing with discrete options, has also been modeled and applied to a number of different games involving simultaneous decisions including paper-rock-scissors (West & Lebiere, 2001) and baseball (Lebiere et al., 2003). The model represents each choice made by the opponent in its given context. That context often includes the sequence of previous choices, bringing in the temporal aspects that will be the focus of additional TACTICS tasks. The model works by storing those decisions in their initial context, then matching against them using the current context. This generates the most likely expected move by the opponent. The player then selects the best move at its disposal to counter that expected move. Our current TACTICS model aims to unify these two paradigms.

The second approach will be used to generate expectations of an attack by the Red player. This will be performed by representing in a single frame the various layers of information to be considered and the outcome for the Red attack in a 0-1 encoding, with 0 meaning no attack and 1 meaning an attack took place. Blended retrieval can then be used to interpolate between those two outcomes, weighing similarity between the current information layers and the past instances stored, and then generating the probability of attack by Red in the current situation. The first approach can then be used to generate the expectation of each move's payoff for the model by combining the probability of Red attack with each model choice to yield an expected payoff, using frames associating each player's choice with the resulting outcome. This approach, together with the re-used Challenge task functionality, can be used to generate all the responses required in the TACTICS tasks.

We will now describe the model structure (see Figure 4) and implementation of each step.

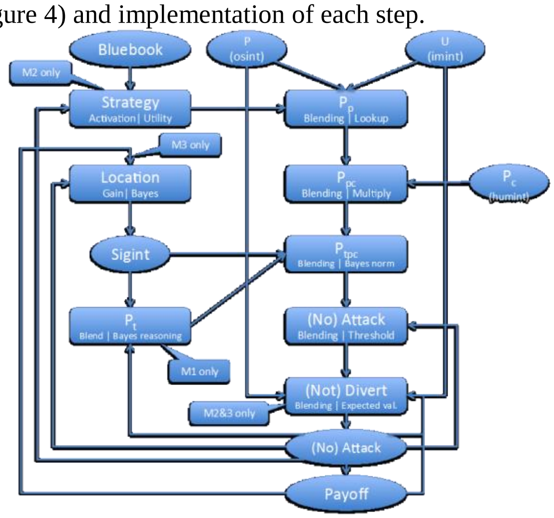

Figure 4. Overview of model structure for TACTICS.

To determine the likely opponent strategy (the first step in task 2), the model uses subsymbolic activation as an estimate of the relative probability of the Bluebook options. Specifically, each Bluebook is represented as a chunk containing its name (e.g., passive or aggressive). Additionally, each option is represented by a set of four chunks separately binding the option name with each propensity value. For each option, the base-level activation for the corresponding chunk provides support for accumulation of evidence through its frequency summation term and for change detection through its power law decay-based recency. Prediction of the opponent strategy is accomplished by retrieving the most active option chunk. Credit assignment is performed by reinforcing the option most likely to have been responsible for the observed outcome. In this case, that means the obvious heuristic of reinforcing the chunk whose probability is closest to the outcome, i.e., passive if no attack occurs and aggressive if an attack occurs.

To compute P_p, the model represents the identity of the player (neutral in task 1, passive and aggressive in task 2), the OSINT value, the IMINT value, and P_p. We train the model directly from the matrix provided in the task instructions, with one chunk for each matrix cell. We defined the OSINT values in the matrix as 0.2 and 0.3 for values less than and greater than 0.25, respectively, and the IMINT values as 2 and 5, respectively. This approach provides similar results to a categorical approach, but in a simpler way with fewer degrees of freedom.

The representation of $P_{p,c}$ includes the value of P_p generated previously, P_c received as a real value [0, 1], and $P_{p,c}$. The model is trained directly from averaging examples using a coarse increment of 20% from 0-100%.

The representation of P_t is currently a representation of SIGINT (as symbolic sigint/no-sigint chunks standing for chatter or silence, respectively), a representation of an attack (as symbolic attack/no-attack chunks), and P_t. Training occurs directly from instructions, with each chunk representing one cell of the conditional probability matrix given to the participants, as similar to the Bluebook matrix (see Figure 3). The model performs a blended retrieval for P_t, specifying the current value of SIGINT and a positive attack value (i.e., attack). This represents a common confusion between opposite conditional probabilities: the model is asked to produce the conditional probability of attack given SIGINT, and accesses the closest thing it has, i.e., the conditional probability of SIGINT given attack.

The generation of $P_{t,p,c}$ estimates is handled identically to $P_{p,c}$, only now the factors include $P_{p,c}$, P_t, and $P_{t,p,c}$. Training and representations are all identical to $P_{p,c}$, and could indeed be handled using the same chunk types, although they currently use separate chunk types.

For all of these functions, new chunks representing the problem solutions are learned at each trial and complement the initial instructions or background knowledge. Base-level learning is turned on and set to its usual decay value of 0.5 to capture effects of recency and frequency. This plays a particularly significant role in the generation of P_t, as well as the identification of the red strategy (passive or aggressive) in Mission 2.

A modeling choice was whether to generate a divert decision directly from $P_{t,p,c}$, P, and U; or to break it down into two simpler stages. The first stage involves generating an expectation of whether red will attack or not based on $P_{t,p,c}$ (and past feedback), while the second stage involves deciding whether to divert or not based on the attack expectation and factors P and U (and past feedback). The advantage of this two-stage retrieval is to make the simpler decisions easier to apply than the complex calculation of expectations from $P_{t,p,c}$, P, and U; and faster to learn by breaking down the representational space into two distinct parts of lower dimensionality

A major issue in modeling was dealing effectively with delayed feedback. The information about an attack from Red and the resulting payoff was available long after many intermediate decisions leading to the divert decision being made. We included the attack and payoff information in the divert decision chunk, making them available to estimate outcomes for each course of action. Propagating that feedback information to the immediately preceding step of generating an attack expectation is fairly easy by keeping that chunk in a buffer such as the imaginal buffer, assuming that subjects maintain that information for a short amount of time.

TACTICS Model Fits

The TACTICS model fit the preliminary human dataset quite well, with an average performance similar to that of the ICArUS Challenge Tasks model. The following graphs (Figure 6) show the fits to human data for each decision stage. Due to the limited number of trials, instead of reporting regression fits, we adopted an RMSD ratio to determine the degree to which the model captures participants' trial-by-trial deviation from rational behavior:

1 – (RMSD Human-Model / RMSD Human-Rational)

The average RMSD ratio across decision points in the two tasks (P_p, $P_{p,c}$, $P_{t,p,c}$) was .735, which is consistent with the fits reported in the ICArUS task.

There are two interesting phenomena. The first is fact that participants, on average, tended to switch strategy in trials 7-8, despite having ample evidence (trials 1-5 were all attacks from red, but trials 6-7 were no attack) that red was an aggressive player, and they were instructed that red did not change strategy during the task. Our model implicitly captured this behavior using only the sub-symbolic activation resulting from reinforcing the aggressive strategy when there was an attack, and reinforcing the passive strategy when there was no attack. The model was not altered or trained in any way to capture human's strategy selection.

The other interesting phenomenon was the pervasive averaging heuristic that occurred, despite task instructions leading participants to generate joint probabilities. In essence, participants were treating subsequent layers of information as independent when task instructions were to treat the layers as dependent.

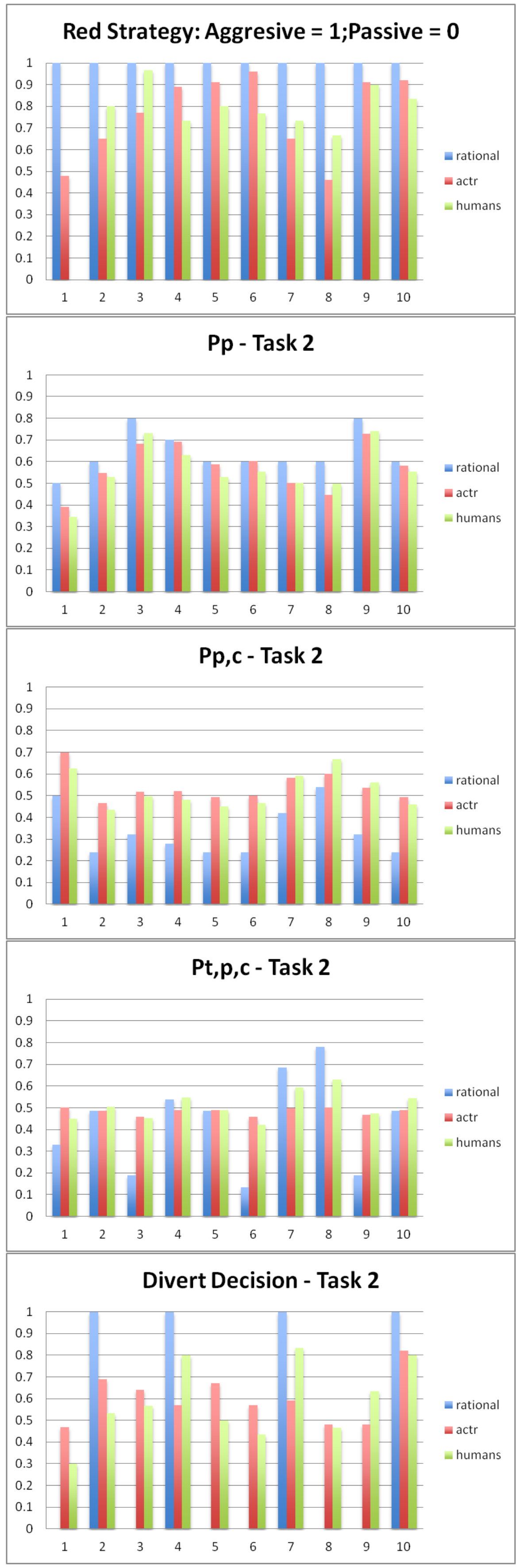

Figure 5. Average model fits to participants for Task 2.

While it was possible that participants were simply misunderstanding task instructions, the pervasiveness of the averaging heuristic (29 of 30 participants exhibited it consistently) indicated that participants were biased to treat sequential probabilities as independent, which may have been due to the complexity of the task environment.

While the current model, which was extended from the ICArUS Challenge Tasks, similarly captures human behavior, the model still has some complexities which limit its adoption as a more general model of sensemaking. To this point, we wish to preview efforts to generalize the instance-based learning framework in a simple general model of sensemaking.

General Sensemaking Model

ACT-R has recently been used to model a task that involved parsing a series of decision-trees (Lebiere Jentsch & Ososky, 2013) across several task scenarios. This model was extended to include a base of seven productions (see Figure 6) which was able to parse any information that may be described in a decision-tree form (i.e., an acyclic directed graph). That said, there is nothing about the control of these productions which precludes cyclic behavior, meaning that the model can theoretically perform any decision whose steps can be broken down into a directed graph.

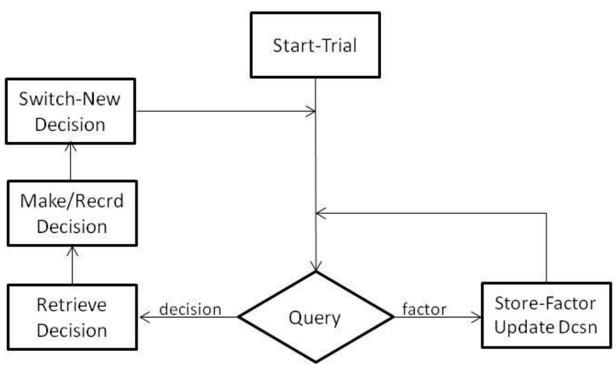

Figure 6. Example of 7 productions capable of general sensemaking processes.

This general model commits to general decision logic and thus can theoretically capture more behavior than a task-specific model. Also, because the model is designed in a cognitive architecture, it is more robust than typical decision trees. This is due to the fact that the model is able to learn from experience and is able to generalize to non-binary and cyclic outputs.

The model takes a series of instructions, called decision-factors, and accesses (i.e., retrieves) factor-values – which are the atomic components of decisions – for the current situation until it is instructed to make a decision. This instruction may be external (i.e., task from instruction) or when a given threshold is reached. At that time, a decision chunk is retrieved either via standard retrievals with partial matching, or by blended retrievals through instance-based learning. The decision is also stored in a factor-value chunk for use in later decision-making steps. With the decision retrieved, the model moves onto the next decision in the chain.

There are three chunk types used in the model. The first is the decision-factor chunk type, which has three slots: type, factor, and index. The type slot determines the type of the decision, the factor slot determines the current factor to be processed, and the index slot records the prior factor that was retrieved (to chain decision-factors together). Decision-factor chunks act as a means of chaining through the elements that go into a decision choosing a factor from a decision-factor chunk, retrieving the factor value (either through sensory input or from memory), and updating the current decision goal.

The second chunk type is the *factor* chunk, which also has three slots: *scenario*, *name*, and *value*. The *scenario* slot holds the name of the current trial, the *name* slot holds the kind of factor (e.g., vulnerability), and the *value* slot holds the value of the factor (e.g., yes/no or numeric). The final chunk type is an *intermediate-decision* chunk which stores the factor chunks from the decision (e.g., slots for vulnerability and opportunity and propensity in the propensity intermediate-decision chunk). These intermediate-decision chunks are effectively frames, with learning across frames occurring due to the blended retrieval mechanism of ACT-R which implements the same instance-based learning theory from the ICArUS Challenge Tasks.

The final chunk type is an *intermediate-decision* chunk which stores the factor chunks from the decision (e.g., slots for vulnerability and opportunity and propensity in the propensity intermediate-decision chunk). The different variants of intermediate-decision chunks do not need to be pre-specified, but may be derived from experience using special P* productions. P* productions allow for slot names to be variabilized, and when the model is provided with a variabilized slot name that does not occur in the specification of the chunk-type, it extends the chunk-type with an extra slot. These intermediate-decision chunks are effectively frames, with learning across frames occurring due to the blended retrieval mechanism of ACT-R which implements the same instance-based learning theory found in ICArUS and TACTICS. Thus a fundamental new capability of the general sensemaking model is the ability to not only learn new frames but also to learn *new frame types* from experience.

While the sequence of decision chains has to be provided as input to the model, they are derived from task instructions. The benefit of the general sensemaking model is that it can process any arbitrary set of instructions using the same core productions. The model does not need to be changed in any way to tackle a new task but rather just needs a new set of instructions, just as human subjects do. In addition, it is possible to perform foraging behavior by having a pre-decision trigger that determines a value (such as expected information gain; EIG) that must reach a given threshold before moving onto the generation or revision of a decision.

Acknowledgement

This work was conducted through collaboration in the Robotics Consortium sponsored by the U.S Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement W911NF-10-2-0016; and by Intelligence Advanced Research Projects Activity via DOI contract number D10PC20021. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DOI, or the U.S. Government.

References

Anderson, J. R. (1990). The Adaptive Character of Thought. Hillsdale, NJ: Erlbaum.

Anderson, J. R., & Betz, J. (2001). A hybrid model of categorization. *Psychonomic Bulletin & Review, 8*, 629-647.

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., Qin, Y. (2004) An integrated theory of Mind. *Psychological Review, 111*, 1036-1060.

Anderson, J. R., and Lebiere, C. (1998), *The atomic components of thought*, Erlbaum, Mahwah, NJ, 1998.

Erev, I., Ert, E., Roth, A. E., Haruvy, E., Herzog, S., Hau, R., Hertwig, R., Stewart, T., West, R., Lebiere, C. (2010). A choice prediction competition, for choices from experience and from description. *Journal of Behavioral Decision Making 23(1)*: 15-47.

Gonzalez, C., Lerch, J. F., & Lebiere, C. (2003). Instance-based learning in dynamic decision making. *Cognitive Science, 27*, 591-635.

Klein, G., Phillips, J. K., Rall, E., & Peluso, D. A. (2006). A data/frame theory of sensemaking. In R. R. Hoffman (Ed.), *Expertise out of context: Proceedings of the 6th International Conference on Naturalistic Decision Making*. Mahwah, NJ: Lawrence Erlbaum & Associates.

Klein, G., Moon, B., & Hoffman, R. (2006). Making Sense of Sensemaking 1: Alternative Perspectives. *Intelligent Systems*, 21 (4), 71.

Lebiere, C. (1999). The dynamics of cognition: An ACT-R model of cognitive arithmetic. *Kognitionswissenschaft.*, 8 (1), 5-19.

Lebiere, C., Pirolli, P., Thomson, R., Paik, J., Rutledge-Taylor, M., Staszewski, J., & Anderson, J. (2013). A Functional Model of Sensemaking in a Neurocognitive Architecture. *Computational Intelligence and Neuroscience*; Special Issue on Neurocognitive Models of Sense Making.

Lebiere, C., Jentsch, F., & Ososky, S. (2013). Cognitive models of decision making processes for Human-Robot Interaction. *Proceedings of the HCI International Conference*, July 2013.

Lebiere, C., Gonzalez, C., & Warwick, W. (2010). Metacognition and multiple strategies in a cognitive model of online control. *Journal of Artificial General Intelligence, 2*(2), 20-37.

Lebiere, C., Gonzalez, C., & Martin, M. (2007). Instance-based decision making model of repeated binary choice. In *proceedings of the 8th International Conference on Cognitive Modeling*. Ann Arbor, Michigan, USA.

Lebiere, C., Gray, R., Salvucci, D. & West R. (2003) Choice and Learning under Uncertainty: A Case Study in Baseball Batting. In *Proceedings of the 25th Annual Meeting of the Cognitive Science Society*. pg 704-709.

Lebiere, C., Anderson, J. R., & Reder, L. M. (1994). Error modeling in the ACT-R production system. In *Proceedings of the Sixteenth Annual Meeting of the Cognitive Science Society*, pp. 555-559. Hillsdale, NJ: Erlbaum.

Rutledge-Taylor, M. F, Lebiere, C., Vinokurov, Y., Stazewski, J., & Anderson, J. R. (2011). Bridging the gap: A neurally plausible functional model of sensemaking. In Samsonovich, A. V., Johannsdottir, K. R., Chella, A., & Goertzel, B. (Eds.). *Proceedings of the Second Annual Meeting of the BICA Society*. Fairfax: IOC Press.

Rutledge-Taylor, M. F, Lebiere, C., Thomson, R., Stazewski, J., & Anderson, J. R. (forthcominng). A Comparison of Rule-Based versus Exemplar-Based Categorization Using the ACT-R Architecture. *The 21st Behavior Representation in Modeling & Simulation Conference.*

Sanner, S., Anderson, J. R., Lebiere, C., & Lovett, M. (2000). Achieving efficient and cognitively plausible learning in backgammon. In *ICML* (pp. 823-830).

Taatgen, N. A., Lebiere, C., & Anderson, J. R. (2006). Modeling paradigms in ACT-R. *Cognition and multi-agent interaction: From cognitive modeling to social simulation*, 29-52.

Tversky, A., & Kahneman, D. (1974). *Judgment under Uncertainty: Heuristics and Biases. Science, 185, 1124-1131*.

Wallach, D. & Lebiere, C. (2003). Conscious and unconscious knowledge: Mapping to the symbolic and subsymbolic levels of a hybrid architecture. In Jimenez, L. (Ed.) Attention and Implicit Learning. Amsterdam, Netherlands: John Benjamin Publishing Company.

West, R. L., & Lebiere, C. (2001). Simple games as dynamic, coupled systems: Randomness and other emergent properties. *Journal of Cognitive Systems Research*, 1(4), 221-239.

Author Biographies

ROBERT THOMSON is research faculty at Carnegie Mellon University in Pittsburgh, PA. His main research interests are learning theories and their application to computational cognitive architectures.

CHRISTIAN LEBIERE is research faculty in the Psychology Department at Carnegie Mellon University. His main research interests are computational cognitive architectures and their applications to psychology, artificial intelligence, human-computer interaction, decision-making, intelligent agents, cognitive robotics and neuromorphic engineering.

STEFANO BENNATI is a researcher at Carnegie Mellon University in Pittsburgh, PA.