Cognitive modeling, symbolic

Lewis, R.L. (1999). Cognitive modeling, symbolic. In Wilson, R. and Keil, F. (eds.), *The MIT Encyclopedia of the Cognitive Sciences*. Cambridge, MA: MIT Press.

Symbolic cognitive models are theories of human cognition that take the form of working computer programs. A cognitive model is intended to be an explanation of how some aspect of cognition is accomplished by a set of primitive computational processes. A model performs a specific cognitive task or class of tasks and produces behavior that constitutes a set of predictions that can be compared to data from human performance. Task domains that have received considerable attention include problem solving, language comprehension, memory tasks, and human-device interaction.

The scientific questions cognitive modeling seeks to answer belong to *cognitive psychology*, and the computational techniques are often drawn from *artificial intelligence*. Cognitive modeling differs from other forms of theorizing in psychology in its focus on functionality and computational completeness. Cognitive modeling produces both a theory of human behavior on a task and a computational artifact that performs the task.

The theoretical foundation of cognitive modeling is the idea that cognition is a kind of COMPUTATION (see also COMPUTATIONAL THEORY OF MIND). The claim is that what the mind does, in part, is perform cognitive tasks by computing. (The claim is not that the computer is a metaphor for the mind, or that the architectures of modern digital computers can give us insights into human mental architecture.) If this is the case, then it must be possible to explain cognition as a dynamic unfolding of computational processes. A cognitive model cast as a computer program is a precise description of what those processes are and how they develop over time to realize some task.

A cognitive model is considered to be a *symbolic* cognitive model if it has the properties of a symbolic system in the technical sense of Newell and Simon's (1976) Physical Symbol System Hypothesis (PSSH). The PSSH provides a hypothesis about the necessary and sufficient conditions for a physical system to realize intelligence. It is a reformulation of Turing computation (see CHURCH-TURING THESIS) that identifies symbol processing as the key requirement for complex cognition. The requirement is that the system be capable of manipulating and composing symbols and symbol structures—physical patterns with associated processes that give the patterns the power to denote either external entities or other internal symbol structures (Newell, 1980; Newell, 1990; Pylyshyn, 1989; Simon, 1996). One of the distinguishing characteristics of symbols systems is that novel structures may be composed and interpreted, including structures that denote executable processes.

The extent to which symbolic processing is required for explaining cognition, and the extent to which *connectionist* (see COGNITIVE MODELING, CONNECTIONIST) models have symbolic properties, has been the topic of ongoing debates in cognitive science (Fodor & Pylyshyn, 1988; Rumelhart, 1989; Simon, 1996). Much of the debate has turned on the question of whether or not particular connectionist systems are able to compose and interpret novel structures. In particular, Fodor & Pylyshyn argue that any valid cognitive theory must have the properties of productivity and systematicity. Productivity refers to the ability to produce and entertain an unbounded set of novel propositions with finite means. Systematicity is most easily seen in linguistic processing, and refers to the intrinsic connection between our ability to produce or comprehend certain linguistic forms. For example, no speaker of English can understand the utterance "John loves the girl" without also being able to understand "the girl loves John", or any other utterance from the unbounded set of utterances of the form "X loves Y". Both productivity and systematicity point to the need to posit underlying abstract structures that can be freely composed, instantiated with novel items, and interpreted on the basis of their structure.

A variety of empirical constraints may be brought to bear on cognitive models. These include: basic *functionality* requirements (a model must actually perform the task to some approximation if it is to be veridical); data from *verbal protocols* of human subjects thinking aloud while PROBLEM SOLVING (these reveal intermediate cognitive steps that may be aligned with the model's behavior (Newell & Simon, 1972; Ericsson & Simon, 1980)); *chronometric data* (such data can constrain a cognitive model once assumptions are made about the time course of the component computational processes (Newell, 1990)); *eye movement* data (eye fixation durations are a function of cognitive, as well as perceptual, complexity; Carpenter & Just, 1987; Rayner 1977); *error patterns*; and data on *learning rates* and *transfer of cognitive skill* (such data constrain the increasing number of cognitive models that are able to change behavior over time; Singley & Anderson, 1990).

Though the problem of under-constraining data is a universal issue in science, it is sometimes thought to be particularly acute in computational cognitive modeling, despite the variety of empirical constraints described above. There are two related sides to the problem. First, cognitive models are often seen as making many detailed commitments about aspects of processing for which no data distinguishes among alternatives. Second, because of the universality of computational frameworks (see CHURCH-TURING THESIS), an infinite number of programs can be created that mimic the desired behavior (Anderson, 1978).

Theorists have responded to these problems in a variety of ways. One way is to adopt different levels of abstraction in the theoretical statements: in short, not all the details of the computer model are part of the theory. Newell (Newell 1990; Newell et al., 1991), Marr (1982), Pylyshyn (1984) and others have developed frameworks for specifying systems at multiple levels of abstraction. The weakest possible correspondence between a model and human cognition is at the level of input/output only: if the model only responds to functionality constraints, it is intended only as a sufficiency demonstration and formal task definition (Pylyshyn, 1984; Pylyshyn, 1989). The strongest kind of correspondence requires that the model execute the same algorithm---take the same intermediate computational steps---as human processing. (No theoretical interpretation of a cognitive model, not even the strongest, depends on the hardware details of the host machine.)

An important method for precisely specifying the intended level of abstraction is the use of programming languages designed for cognitive modeling, such as PRODUCTION SYSTEMS. Production systems were introduced by Newell (1973) as a flexible model of the control structure of human cognition. The flow of processing is not controlled by a fixed program or procedure laid out in advance, as is the case in standard procedural programming languages. Instead, production systems posit a set of independent production rules (condition-action pairs) that may fire anytime their conditions are satisfied. The flow of control is therefore determined at run-time, and is a function of the dynamically evolving contents of the working memory that triggers the productions. A cognitive model written in a production system makes theoretical commitments at the level of the production rules, and defines a computationally complete system at that level. The particular underlying implementation (e.g., LISP or Java) is theoretically irrelevant.

A complementary approach to reducing theoretical degrees of freedom is to apply the same model with minimal variation to a wide range of tasks. Each new task is not an unrelated pool of data to be arbitrarily fitted with a new model or with new parameters. For example, a computational model of short-term memory that accounts for immediate serial recall should also apply, with minimal strategy variations, to free recall tasks and recognition tasks as well (Anderson & Matessa, 1997).

Recent cognitive modeling research combines these approaches by building and working with COGNITIVE ARCHITECTURES. A cognitive architecture posits a fixed set of computational mechanisms and resources that putatively underlie a wide range of human cognition. Since these cognitive architectures never correspond to the architectures of modern computers (for example, they may demand a higher degree of parallelism), the architectures must first be *emulated* on computers before cognitive models can be built within them for specific tasks. Such architectures, together with the variety of empirical constraints outlined above, place considerable constraint on task models.

Examples of the architectural approach include ACT-R (Anderson, 1993), CAPS (Just & Carpenter, 1992), Soar (Newell, 1990), EPAM (Feigenbaum & Simon, 1984), and Epic (Meyer & Kieras, 1997) (all are production systems, with the exception of EPAM). These architectures have collectively been applied to a broad set of phenomena in cognitive psychology. For example, Anderson and colleagues (Anderson, 1993, Singley & Anderson. 1989) have demonstrated that a production rule analysis of cognitive skill, along with the learning mechanisms posited in the ACT architecture, provide detailed and explanatory accounts of a range of regularities in cognitive skill acquisition in complex domains such as learning to program Lisp. ACT also provides accounts of many phenomena surrounding the recognition and recall of verbal material (e.g., the fan effect), and regularities in problem solving strategies (Anderson, 1993; Anderson & Lebiere, 1998). EPAM is one of the earliest computational models in psychology and accounts for a significant body of data in the learning and high-level perception of verbal material. It has been compared in some detail to related connectionist accounts (Richman & Simon, 1989). Soar is a learning architecture that has been applied to domains ranging from rapid, immediate tasks such as typing and video game interaction (John, Vera & Newell, 1994) to long stretches of problem solving behavior (Newell, 1990), building on the earlier analyses by Newell & Simon (1972). Soar has also served as the foundation for a detailed theory of sentence processing, which models both the rapid on-line effects of semantics and context, as well as subtle effects of syntactic structure on processing difficulty across several typologically distinct languages (Lewis, 1996; Lewis, in preparation). EPIC is a recent architecture that combines a parallel production system with models of peripheral and motor components, and accounts for a substantial body of data in the performance of dual cognitive tasks (Meyer & Kieras, 1997). CAPS is a good example of recent efforts in symbolic modeling to account for individual differences in cognitive behavior. CAPS explains differences in language comprehension performance by appeal to differences in working memory capacity (Just & Carpenter, 1992). Polk & Newell (1995) developed a constrained parametric model of individual differences in syllogistic reasoning that provides close fits to particular individuals by making different assumptions about the way they interpret certain linguistic forms (see also Johnson-Laird & Byrne, 1992).

In short, modern symbolic cognitive modeling is characterized by detailed accounts of chronometric data and error patterns; explorations of the explanatory role of the same basic architectural components across a range of cognitive tasks; attempts to clearly distinguish the contributions of relatively fixed architecture and more plastic task strategies and background knowledge; and attempts to explicitly deal with the problem of theoretical degrees of freedom. The underlying goal of all these approaches is to produce more unified accounts of cognition explicitly embodied in computational mechanisms.

Anderson, J.R. (1978). Arguments concerning representations for mental imagery. *Psychological Review*, 85:249-277.

Anderson, J.R. (1993). *The Adaptive Character of Thought*. Hillsdale, NJ: Erlbaum.

Anderson, J.R. and Matessa, M. (1997). A production system theory of serial memory. *Psychological Review*, 104(4):728–748.

Anderson, J.R. and Lebiere, C. (1998) *The Atomic Components of Thought.* Hillsdale, NJ: Lawrence Erlbaum.

Ericcson, K.A. and Simon, H.A. (1984). *Protocol analysis: Verbal reports as dat*a. Cambridge, MA: MIT Press.

Feigenbaum, E. and Simon, H.A. (1984). EPAM-like Models of Recognition and Learning, *Cognitive Science*, 8:305-336.

Fodor, J.A. and Pylyshyn, Z.W. (1988) Connectionism and cognitive architecture: A critical analysis. *Cognition* 28:3-71.

John, B.E., Vera, A.H., and Newell, A. (1994). Toward real-time GOMS: A model of expert behavior in a highly interactive task. *Behavior and Information Technology*, 13:255-267.

Just, M.A. and Carpenter, P.A (1987) *The Psychology of Reading and Language Comprehension*. Boston: Allyn and Bacon.

Just, M.A. and Carpenter, P.A. (1992). A capacity theory of comprehension: Individual differences in working memory. *Psychological Review*, 99(1):122-149.

Lewis, R.L. (1996) Interference in short-term memory: The magical number two (or three) in sentence processing. *Journal of Psycholinguistic Research* 25:93-115.

Lewis, R.L. (in preparation). *Cognitive and Computational Foundations of Sentence Processing*. Oxford: Oxford University Press.

Meyer, D. E. and Kieras, D. E. (1997). A computational theory of executive cognitive processes and multiple-task performance: Part 1. Basic mechanisms. *Psychological Review*, 104(1):3-65.

Marr, D. (1982). Vision. New York: Freeman.

Polk, T.A. and Newell, A. (1995). Deduction as verbal reasoning. *Psychological Review*, 102(3): 533-566.

Pylyshyn, Z.W. (1984). *Computation and Cognition*. Cambridge, MA: Bradford/MIT Press.

Pylyshyn, Z.W. (1989). Computing in cognitive science. In Posner, M.I. (ed.), *Foundations of Cognitive Science*. Cambridge, MA: MIT Press.

Richman, H.B. and Simon, H.A. (1989). Context effects in letter perception: Comparison of two theories. *Psychological Review*, 96(3):417-432.

Newell, A. and Simon, H. A. (1976) Computer science as empirical inquiry: Symbols and Search. *Communications of the ACM*, 19:113-126.

Newell, A. (1980) Physical symbol systems. *Cognitive Science*, 4:135-183.

Newell, A. (1973). Production systems: Models of control structures. In W.G. Chase (Ed.), *Visual Information Processing* (pp. 463-526). San Diego, CA: Academic Press.

Newell, A. (1990) *Unified Theories of Cognition*. Cambridge, MA: Harvard.

Newell, A. and Simon, H.A. (1972) *Human Problem Solving*. Englewood Cliffs, NJ: Prentice-Hall.

Newell, A., Yost, G. Laird, J.E., Rosenbloom, P.S. and Altmann, E. (1991).

Formulating the problem-space computational model. In Rashid, R.F. (ed.), *CMU Computer Science: A 25th Anniversary Commemorative*. Reading, MA: Addison-Wesley.

Rayner, K. (1977). Visual attention in reading: Eye movements reflect cognitive processes. *Memory & Cognition* 4:443-448.

Rumelhart, D.E. (1989). The architecture of mind: A connectionist approach. In Posner, M.I. (ed.), *Foundations of Cognitive Science*. Cambridge, MA: MIT Press.

Simon, H.A. (1996). The patterned matter that is mind. In Steier, D. and Mitchell, T. (eds.), *Mind Matters: Contributions to Cognitive and Computer Science in Honor of Allen Newell*. Hillsdale, NJ: Erlbaum.

Singley, M.K. and Anderson, J.R. (1989). *The transfer of cognitive skill*. Cambridge, MA: Harvard University Press.