69

Cognitive Modeling

Eric Chown
Bowdoin College

69.1 Introduction

An important goal of cognitive science is to understand human cognition. Good models of cognition can be *predictive* — describing how people are likely to react in different scenarios — as well as *prescriptive* — describing limitations in cognition and potentially ways in which the limitations might be overcome. In a sense, the benefits of having cognitive models are similar to the benefits individuals accrue in building their own internal model. To quote Craik [1943]:

> If the organism carries a 'small-scale model' of external reality and of its own possible actions within it head, it is able to try out various alternatives, conclude which is the best of them, react to future situations before they arise, utilize the knowledge of past events in dealing with the present and future, and in every way to react in a much fuller, safer, and more competent manner to the emergencies which face it. (p. 61)

Among the important questions facing cognitive scientists are how such models are created and how they are represented internally. Craik emphasizes the importance of the predictive power of models, and it is the model's ability to make accurate predictions that is the ultimate measure of the model's value. One important value of computers in cognitive science is that computer simulations provide a means to instantiate theories and to concretely test their predictive power. Further, implementation of a theory in a computer model forces theoreticians to face practical issues that they may never have otherwise considered.

The role of computer science in cognitive modeling is not strictly limited to implementation and testing, however. A core belief of most cognitive scientists is that cognition is a form of computation (otherwise, computer modeling is a doomed enterprise) and the study of computation has long been a source of ideas (and material for debates) for building cognitive models. Computers themselves once served as the dominant metaphor for cognition. In more recent years, the influence of computers has been more in the area of computational paradigms, such as rule-based systems, neural network models, etc.

1-58488-360-X/$0.00+$1.50

69.2 Underlying Principles

One of the dangers of cognitive modeling is falling under the spell of trying to apply computational models directly to cognition. A good example of this is the "mind as computer" metaphor that was once popular but has fallen into disfavor. Computer science offers a range of computational tools designed to solve problems, and it is tempting to apply these tools to psychological data and call the result a cognitive model. As McCloskey [1991] has pointed out, this falls far short of the criteria that could reasonably used to define a theory of cognition. One replacement for the "mind as computer" metaphor makes this point well. Neurally inspired models of cognition fell out of favor following Minsky and Papert's 1969 book *Perceptrons* that showed that the dominant neural models at the time were unable to model nonlinear functions (notably exclusive-or). The extension of these models by the PDP (Parallel Distributed Processing) group in 1986 [Rumelhart and McClelland, 1986] is largely responsible for the **connectionist** revolution of the past 25 years. The excitement generated by these models was twofold: (1) they were computationally powerful and simple to use, and (2) as neural-level models they appeared to be physiologically plausible.

A major difficulty for connectionist theory of the past 20 years has been that, despite the fact that the early PDP-style models (particularly models built upon **feed-forward back-propagation** networks) were proven to be implausible for both physiological and theoretical reasons (e.g., see [Lachter and Bever, 1988; Newell, 1990]), many cognitive models are still built using such discredited computational engines. The reason for this appears to be simple convenience. Back-propagation networks, for example, can approximate virtually any function and are simple to train. Because any set of psychological data can be viewed as a function that maps an input to a behavior, and because feed-forward back-propagation networks can approximate virtually any function, it is hardly surprising that such networks can "model" an extraordinary range of psychological phenomena. To put this another way, many cognitive models are written in computer languages like C. Although such models may accurately characterize a huge range of data on human cognition, no one would argue that the C programming language is a realistic model of human cognition. Feed-forward neural networks seem to be a better candidate for a cognitive model because of some of their features: they intrinsically learn, they process information in a manner reminiscent of neurons, etc. In any regard, this suggests that the criteria for judging the merits of a cognitive model must include many more constraints than whether or not the model is capable of accounting for a given data set. While issues such as how information is processed are useful for judging models, they are also crucial for constructing modals.

There are a number of sources and types of constraints used in cognitive modeling. These break down relatively well by the disciplines that comprise the field. In practice most cognitive models draw constraints from some, but not all, of these disciplines. In broad terms, the data for cognitive models comes from psychology. "Hardware" constraints come from neuroscience. "Software" constraints come from computer science, which also provides methodologies for validation and testing. Two related fields that are relatively new, and therefore tend to provide softer constraints are evolutionary psychology and environmental psychology. The root idea of each of these fields is that the evolution process, and especially the environmental conditions that took place during evolution, are crucially important to the kind of brain that we now have. We will examine the impact of each field on cognitive modeling in turn.

69.2.1 Psychology

The ultimate test of any theory is whether or not it can account for, or correctly predict, human behavior. Psychology as a field is responsible for the vast majority of data on human behavior. Over the last century the source of this data has evolved from mere introspection on the part of theorists to rigorous laboratory experimentation. Normally the goal of psychological experiments is to isolate a particular cognitive factor; for example, the number of items a person can hold in short term memory. In general this isolation is used as a means of reducing complexity. In principle this means that cognitive theories can be constructed piecewise instead of out of whole cloth. It would be fair to say that the majority of work in cognitive science proceeds on this principle. A fairly typical paper in a cognitive science conference proceeding, for example,

will present a set of psychological experiments on some specific area of cognition, a model to account for the data, and computer simulations of the model.

69.2.2 Neuroscience

The impact of neuroscience on cognitive science has grown dramatically in conjunction with the influence of neural models in the last 20 years. Unfortunately, terms such as "neurally plausible" have been applied fairly haphazardly in order to lend an air of credibility to models. In response, some critics have argued that neurons are not well understood enough to be productively used as part of cognitive theory. Nevertheless, though the low level details are still being studied, neuroscience can provide a rich source of constraints and information for cognitive modelers. Within the field there are several different types of architectural constraint available. These include:

1) *Information flow*. We have learned from neuroscientists, for example, that the visual system is divided into two distinct parts, a "what" system for object identification, and a "where" system for determining spatial locations. This suggests computational models of vision should have similar properties. Further, these constraints can be used to drive cognitive theory as with the PLAN model of human cognitive mapping [Chown, et al. 1995]. In PLAN it was posited that humans navigate in two distinct ways, each corresponding to one of the visual pathways. Virtually all theories of cognitive mapping had previously included a "what" component based upon topological collections of landmarks, but none had a good theory of how more metric representations are constructed. The split in the visual system led the developers of PLAN to theorize that metric representations would have simple "where" objects as their basic units. This led directly to a representation built out of "scenes," which are roughly akin to snapshots.

2) *Modularity*. A great deal of work in neuroscience goes towards understanding what kinds of processing is done by particular areas of the brain, such as the hippocampus. These studies can range from working with patients with brain damage to intentionally lesioning animal brains. More recently, imaging techniques such as fMRI (functional magnetic resonance imaging) have been used to gain information non-invasively. This work has provided a picture of the brain far more complex than the simple "right brain-left brain" distinction of popular psychology. The hippocampus, for example, has been implicated in the retrieval of long-term memories [Squire, 1992] as well as in the processing of spatial information [O'Keefe and Nadel, 1978]. In principle, discovering what each of the brain's different subsystems does is akin to determining what each function that makes up a computer program does.

Modularity in the brain, however, is not as clean as modularity in computer programs. This is largely due to the way information is processed in the brain, namely by neurons passing activity to each other in a massively parallel fashion. Items processed close together in the brain, for example, tend to interfere with each other because neural cells often have a kind of inhibitory surround. This fact is useful in understanding how certain perceptual processes work. Further it means that when one is thinking about a certain kind of math problem, it may be possible to also think about something unrelated like what will be for dinner, but it will be more difficult to simultaneously think about another math problem. The increased interference between similar items (processed close together) over items processed far apart has been called "the functional distance principle" by Kinsbourne [1982]. This suggests, among other things, that there may not be a clean separation of "modules" in the brain, and further that even within a module architectural issues impact processing.

3) *Mechanisms*. Numerous data are simpler to make sense of in the context of neural processing mechanisms. A good example of this would be the **Necker cube**. From a pure information processing point of view, there is no reason that people would only be able to hold one view of the cube in their mind at a time. From a neural point of view, on the other hand, the perception of the cube can be seen as a competitive process with two mutually inhibitory outcomes. Perceptual theory is an area that has particularly benefited from a neural viewpoint.

4) *Timing*. Perhaps the most famous constraint on cognitive processing offered by neuroscience is the "100 step rule." This rule is based upon looking at timing data of perception and the firing rate of

neurons. From these it has been determined that no perceptual algorithm could be more than 100 steps long (though the algorithm could be massively parallel as the brain itself is).

69.2.3 Computer Science

Aside from providing the means to implement and simulate models of cognition, computer science has also provided constraints on models through limits drawn from the theory of computation, and has been a source of algorithms for modelers.

One of the biggest debates in the cognitive modeling community is whether or not computers are even capable of modeling human intelligence. Critics, normally philosophers, point to the limitations on what is computable and have gone as far as suggesting that the mind may not be computational. While some find these debates interesting, they do not actually have a significant impact on the enterprise of modeling. On the other hand, there have been theoretical results from computational theory that have had a huge impact on the development of cognitive models. Probably the best example of this is the previously mentioned work done by Minsky and Papert on Perceptrons [1969]. They showed that perceptrons, which are a simple kind of neural network, are not capable of modeling nonlinear functions (including exclusive-or). This result effectively ended the majority of neural network research for more than a decade until the PDP group developed far more powerful neural network models [Rumelhart and McClelland, 1986].

69.2.4 Evolutionary and Environmental Psychology

In recent years two branches of psychology have come to prominence as providing alternate sources of constraints based upon evolution, and in particular the kinds of environments in which humans evolved. Evolutionary psychology is most often associated with the work of Tooby and Cosmides (e.g. [Tooby and Cosmides, 1992]) while environmental psychology is often associated with the work of Steve and Rachel Kaplan (e.g. [Kaplan & Kaplan, 1989]). What both of these fields have in common is a belief that the brain should not be studied in a vacuum, that some types of context are extremely meaningful.

In the case of evolutionary psychology the context is provided by evolution. As has been noted in many places, systems that are evolved rather than designed, tend to end up looking like the work of a "tinkerer." The eye is a well known example of a system that is poorly "designed" but is nonetheless functional and can be understood as a series of successive improvements, each adding functionality to the previous iteration [Dawkins, 1986]. This example captures the core tenets of evolutionary psychology, that evolution tends to work in piecemeal fashion with each change adding functionality to what existed previously. This does not tend to be a hard constraint on cognitive models, as it can be argued that the evolutionary story behind any particular theory simply has yet to be found. Nevertheless, the evolutionary view provides a powerful way to think about how pieces of the cognitive system came about and for what purpose.

As the name would suggest, work in environmental psychology focuses on the environment as a source of constraints. Most environmental psychologists focus their research on how people interact with different kinds of environments and how to use this knowledge to design better spaces. Another branch of the field, however, has noted that the environment adds additional meaningful context to evolutionary history. The evolutionary history of the brain is a story of information processing mechanisms that evolved to address the specific needs of our ancestors. The human ability to represent and reason about large-scale space, for example, allowed our ancestors to forage and hunt over large areas of savanna. In turn once these spatial abilities were in place they were available for the greater cognitive system and impact cognition of virtually every type [Chown, 1999]. The importance in understanding the evolutionary environments that the brain developed in is highlighted by the work of the Kaplans and their colleagues. The Kaplans have shown, for example, that people will recover more quickly in hospitals with views of nature, perform better in workplaces with views of trees, etc (for reviews see [Kaplan, 1993; Kaplan and Peterson, 1993]). This is a clear indicator that people do not treat information neutrally. As Kaplan and others [Chown, et al. 2002] have argued, the human emotional system can be understood in these terms. The argument is based upon the idea that human emotions address the need our ancestors had to make very fast decisions in encounters

with other dangerous predators. In such cases it is usually better to act quickly than to pause and consider an optimal strategy. This view of cognition undercuts rationality approaches and helps explain many of the supposed shortcomings in human reasoning.

Some of the advantages of an evolutionary/environmental perspective have been clarified by work in robotics. Early artificial intelligence and cognitive mapping focused on reasoning, for example. This led to models that were too abstract to be implemented on actual robots. The move to using robots forced researchers to come at the problem from a far more practical point of view and to consider perceptual issues more directly.

69.3 Research Issues

Since so much about cognition is still not well understood, this section will focus on two of the key debates driving research in the field. These include: 1) is the brain a symbol processor, or does it need to be modeled in neural terms? 2) Should the field be working on grand theories of cognition, or is it better to proceed on a reductionist path?

69.3.1 Are We Symbol Processors?

The connectionist revolution brought a new way of thinking to cognitive science. The critical idea is rather simple-since the "hardware" of the brain is neural, then models of the brain should be described in neural terms. Lending credence to this position was a series of neural models that had a number of attractive properties that seemed notably lacking in symbolic models of the time (e.g. **content addressable memory**, **graceful degradation**, etc. [Rumelhart and McClelland, 1986]). On the face of it, the argument for neural models seems unassailable given that the brain is a neural system. Symbolists, notably Newell [1990] and Fodor and Plyshen [1988] have attacked these models on a number of grounds, however. Their arguments are based upon the idea that the brain, like any complex system, is hierarchical. Neural models, so the argument goes, provide appropriate computational descriptions for only the lowest levels of the cognitive hierarchy. From the point of view of the symbolists, these levels of cognition are also less well understood and not as interesting from a behavioral point of view as the so-called **cognitive band** [Newell, 1990]. From this point of view the operation of the cognitive band is nothing like a neural network, but is much more like a traditional symbol system. The argument is akin to finding the appropriate level at which to study computers. It is possible, and often necessary, to look at the performance of a computer from the point of view of gates. When trying to understand the performance of a complicated piece of software, however, it is much more appropriate to study the performance at the level of a high level programming language. Further, symbolists have effectively argued that current connectionist models are not capable of the full range of behaviors needed for cognitive modeling [Newell, 1990].

The argument for symbolic models comes from computational theory. It is based upon the idea of computational equivalence. Since symbolic models are Turing-complete they are equivalent computationally to any other Turing-complete model. Along these lines a number of efforts have been made to implement symbolic models in neural hardware. The case can then be made that this is exactly what the brain does. Symbolists see this equivalence as freeing them from the need to worry about mechanisms. Even so, the impact of connectionism and the capability of neural models for pattern recognition has led even strongly symbolic models like Soar [Laird, et al., 1987] to use neural networks as pattern recognizers to obtain the symbols.

The freeing up from the constraints of mechanisms has been something of a double-edged sword for symbolic models. On the one hand, symbolic models are easily implemented on computers and relatively easy to expand, debug, etc. Further, in terms of high-level behavior, symbolic models have been shown to be capable of a much wider range of behaviors than their current connectionist counterparts. For example, Soar agents are capable of modeling the flying behaviors of combat pilots [Jones et al., 1999]. On the other hand, critics have complained that systems like Soar are little more than symbolic programming languages. As noted earlier, while a computer running C may be Turing-complete, it would be ridiculous to call

it a model of human cognition. Clark [2001] refers to this as "surface mimicry" and points out that the Soar model is far more homogeneous than the "Swiss Army Knife" model suggested by Tooby and Cosmides [1992]. The critics argue that symbolic models like Soar and ACT-R are under constrained, and therefore they cannot truly be called cognitive models. The lack of constraints goes even further than just mechanisms, symbolic models have also been attacked on evolutionary grounds. It is difficult to see how symbolic constructs like distal memory access could have evolved in any sort of piecemeal fashion.

There are several arguments that symbolists use against neural models. First is the "levels of modeling" argument. This argument posits that we simply do not understand the behavior of neurons well enough to construct credible models from them. To be fair, the majority of neural models do not model the behavior of individual neurons. Other attacks on connectionist models are based upon what current models cannot do. Popular connectionist models, such as feed-forward back-propagation networks, for example, exhibit a number of problems as memory models including "catastrophic forgetting" of old material when presented with new material [McCloskey and Cohen, 1989] . While it is certainly appropriate to attack the individual models on these grounds, it is less so to attack the entire connectionist position based on the failure of even its most notable examples. Similarly, other criticisms of connectionism have attacked the models as being nothing more than new versions of the discredited behaviorist position [Lachter and Bever, 1988]. Again, this is certainly the case with many connectionist models, and should rightfully lead to a search for better models, but it cannot undermine the general position.

A more interesting criticism focuses upon the way that connectionists have pursued cognitive modeling. The argument is the same as the one against some of the symbolic programs. As McCloskey [1991] put it, connectionists often pursue a path of "simulation in search of theory" (p. 388). McCloskey argues that many of these simulations are little better than "black boxes." Modelers do not provide insight into what aspects of the network are crucial to its performance with regard to the task. For example, if back-propagation was used to train the network, is that a crucial step, or could another training regime have been used? If back-propagation were crucial then it would undermine the model if back-propagation were found implausible on other grounds. In other words, connectionist systems are engaged in the same sort of "surface mimicry" as symbolic systems under the guise of "neural plausibility." McCloskey goes on to complain that connectionist models generally fall short in describing how their networks elucidate the cognitive processes they purport to model. To put it simply, connectionist networks are not well understood enough to tell us exactly how they manage to accomplish what they are trained for.

So far in discussing connectionist systems, we have avoided discussing connectionist "symbols." While it may be the case that connectionist units represent collections of neurons, they do not normally represent what most computer scientists would think of as symbols. There are, however, connectionist models that recognize the power of symbols as a basic unit of thought. Many of these models trace their lineage to the work of D.O. Hebb's book *The Organization of Behavior* [1949]. Hebb proposed that the "symbols" of thought were cell assemblies, tightly connected groups of neurons capable of functioning as a unit because of their strong interconnections. The problem with cell assemblies as originally formulated by Hebb was that all of the connections between neurons were positive and only became stronger through learning. Hebb omitted inhibition because there was still no hard evidence of it at the time of publication. A later simulation of the cell assembly construct [Rochester, et al., 1956] showed that without inhibition activity in the simulated brain quickly grew out of control. Unfortunately these results were sufficient to essentially stop research on cell assemblies for more than a decade even though the same paper showed that with the addition of inhibition the cell assembly construct was viable. In recent years, however, the cell assembly idea has undergone a revival as researchers from a number of domains have proposed models based upon Hebb's original idea, but modified with modern understandings of neuroscience [Kaplan et al., 1991; Amit, 1995].

Neural models based upon cell assemblies purport to contain the best of both symbolic and connectionist models. It is difficult to study sequence learning, for example, without something approximating a symbol to serve as a unit in the sequence. Further, many connectionist systems do not address temporal issues at all. Conversely, symbolic models do not ground the symbols in any physical mechanism, nor are symbolic systems able to take advantage of the properties of neural hardware. Chown [2002], for

example, has shown that some learning results that have defied conventional modeling for nearly 40 years can be fairly easily explained when basic neural properties are accounted for in a cell assembly-based model.

69.3.2 Grand Theories?

In his book *Unified Theories of Cognition* Allen Newell [1990] called for a return to all-encompassing theories of mind called **UTCs** after the title of the book. Newell's reasoning echoes McCloskey's complaints about connectionism [1991], that by operating at too small a scale cognitive modelers have worked on under-constrained models. While it may be true, for example, that connectionist system X can model psychological data set Y, such models rarely address questions of how they would or could fit into a larger system. Modeling efforts such as these are sometimes attacked on the grounds that they are "doomed to succeed" in much the same way that models with too many parameters can fit all types of curves. Put another way, if a model is Turing complete, the question of whether or not it can be used to fit some data is not particularly interesting. The interesting question is whether or not there is actually evidence for it. McCloskey and others argue that for these reasons theory should drive simulation rather than the other way around.

Ironically, the two most notable examples of UTCs, Soar and ACT-R, have been attacked on virtually the same grounds-that they are under-constrained. Both systems are built on the same assumption, that at higher levels of cognition the brain is a rule-based system. At the heart of each system is a production system implementing the rule-base that serves as long-term memory. Since production systems are Turing-complete they are capable of modeling anything. To be fair, however, Soar does make some key theoretical commitments that can be used to judge its merits. First and foremost, Soar is a symbol manipulation system with all that that implies. Second, in Soar deliberative thought is equated to a search through a problem space. For example, Soar is equipped with all the basic weak search methods including breadth-first, depth-first, etc. At one time another key constraint associated with Soar was that all learning came as the result of a single mechanism called "chunking" [Laird, et al., 1984]. It is not clear, however, from recent work in SOAR that this is still held as a central tenet of the system.

Soar and ACT-R have both been attacked for their commitment to production systems. This criticism actually pre-dates either system and is most famously associated with the philosopher Hubert Dreyfus [1972]. The Dreyfus position is that systems based upon rules are too brittle to account for the richness of human behavior. For example, Dreyfus discusses how knowledge about the health of a jockey's mother might influence how a bettor would make a wager. It seems unlikely that the bettor would have explicit rules dealing with such a situation, and yet humans are capable of dealing with such situations with ease. The response to this criticism has been to test it explicitly, Either with Doug Lenat's CYC [Lenat and Feigenbaum, 1992] which aims to capture enough knowledge to perform common sense reasoning, or with the Soar program which builds more and more complex agents capable of difficult tasks such as flying jet airplanes in combat situations. In part, Dreyfus's criticisms can also be addressed by noting that rules need not all be specified at the same level of generality. For example, while a system for betting on horses might contain many specific rules concerning the records of horses and jockeys, a general cognitive system might reasonably be expected to include rules such as "when something traumatic happens to a person they will not perform at normal levels." Of course this raises further questions of how such rules are learned, how patterns such as "something traumatic" are recognized, etc. Dreyfus would argue that this leads to an endless cycle for any reasonably complex task.

There are also connectionist programs that work at the level of large theories of cognition. Steven Grossberg, for example, has produced a huge body of work that have never been explicitly put forth as a UTC, but which when viewed as a whole have many of the same principles. Probably the best example of this work is the ART model developed in conjunction with Gail Carpenter [1987]. The SESAME group, operating mainly out of the University of Michigan is also working on a cognitive architecture [Kaplan et al., 1991]. The SESAME architecture is based on the cell assembly and is also the only cognitive architecture to include a complete theory of spatial processing [Chown et al., 1995].

69.4 Best Practices

As the previous sections suggest, there are a number of pitfalls involved in putting together a cognitive model. History has shown that there are two problems that crop up again and again. The first is the danger of constructing a simulation without theoretically motivating the details. This is akin to the old saw that "if you have a big enough hammer everything looks like a nail." There is a related danger that once a simulation works (or at least models the data) it is often difficult to say why. Together these dangers suggest that there should be a close relationship between theory and the simulation process. The goal of a simulation should not be simply to model a dataset, but should also be to elucidate the theory. For example, some connectionist models propose a number of mechanisms as being central to understanding a particular process. These models can be systematically "damaged" by disabling the individual mechanisms. In many cases the damage to the model can be equated to damage to individuals. This provides a second dataset to model, and provides solid evidence of what the mechanism does in the simulation. Alternatively models can be built piecewise mechanism by mechanism. Each new piece of the simulation would correspond to a new theoretical mechanism aiming to address some shortcoming of the previous iteration. This motivates each mechanism and helps to clearly delineate its role in the overall simulation. In the SESAME group this style of simulation has been termed "the systematic exploitation of failure" by one of its members, Abraham Kaplan.

One of the earliest examples of this approach was done by Booker [1982] in an influential work that has helped shaped the adaptive systems paradigm. In an adaptive systems paradigm a simple creature is placed in a microworld where the goal is survival. Creatures are successively altered (and sometimes the environments are as well) by adding and subtracting mechanisms. In each case the success of the new mechanism can be judged by improvements in the survival rate of the organism. In addition to providing a way to motivate theoretical mechanisms, this paradigm is also essentially the same one used for the development of genetic algorithms.

The Soar architecture is probably the pre-eminent symbolic cognitive architecture. Soar is based upon a number of crucial premises that constrain all models written in Soar (which can be considered a kind of programming environment). First, Soar is a rule-based system implemented as a production system. In the Soar paradigm the production rules represent long-term memory and knowledge. One effect of a production firing in Soar can be to put new elements into working memory, Soar's version of short-term memory. For example, a Soar system might contain a number of perceptual productions that aim to identify different types of aircraft. When a production fires it might create a structure in working memory to represent the aircraft it identifies. This structure in turn might cause further productions to fire. Soar enforces a kind of hierarchy through the use of a subgoaling system. Productions can be written to apply generally, or might only match when a certain goal is active. The combination of goals and productions forms a problem space that provides the basic framework for any task. Finally, the Soar architecture contains a single mechanism for learning called "chunking." Essentially Soar systems learn when they reach an impasse generally created by not being able to match any productions. When impasses occur Soar can apply weak search methods to the problem space in order to discover what to do. Once a solution is found, a new production or "chunk" is created to apply to the situation.

Here is an example of a Soar production taken from the Soar tutorial [Laird, 2001]. In this example the agent is driving a tank in a battle exercise.

```
1   sp {wander*propose*move
2       (state ¡s¿ ∧ name wander
3           ∧io.input-linked-blocked.forward no)
4   –
5       (¡s¿ ∧operator ¡o¿ + = )
6       (¡o¿ ∧name move
7           ∧ actions.move.direction forward) }
```

In the example "sp" stands for "Soar production" and starts every production. The production is named "wander*propose*move". The elements that come before the arrow represent the "if" part of the production. In this case the production fires only if the current subgoal is to wander and the forward direction is not blocked (input comes from a specialized structure tagged ∧io). The elements that come after the arrow represent the "then" parts of the production. In this case a new working memory element is created to represent the operator for moving forward. In a typical production cycle, productions are matched in parallel and can propose operators such as the move operator in this case. Then other productions can be used to select among the proposed operators. This selection can be based upon virtually any criteria; for example cognitive productions may be selected over more reactive productions. Production matching can be done in parallel to simulate the parallelism of the brain.

Both the Soar and ACT communities are engaged in programs of simulating more and more human behavior. These simulations can be done at the level of models of simple psychological experiments, or, as is increasingly the case, they can simulate human performance on complex tasks such as flying airplanes. The implicit argument is that if they can simulate anything that humans can do then they must be modeling human cognition. On one level this argument has merit, if either architecture can accurately simulate human performance then it can be used in a predictive fashion. Tac-air Soar [Jones et al., 1999], for example, simulates the performance of combat pilots and is used to train new pilots in a more cost-effective way than if experienced pilots had to be used. On the other hand equivalent functionality is not the same thing as equivalence. As noted previously, critics point out that both Soar and Act are essentially Turing-complete programming environments and therefore are capable of simulating any computable function given clever enough programmers. The fact that both systems still rely heavily on clever programming is still a major limitation with regard to being considered a fully realized model of human cognition. Although Soar's initial success was due in large part to its learning mechanism, for example, little progress has been made within the Soar community in building agents that exhibit any sort of developmental patterns. It is much simpler to build a Soar system that can fly planes than one that can learn to fly planes.

Gail Carpenter, Stephen Grosseberg and their associates have also attacked a wide range of problems, but have done so with much more of an eye towards cognitive theory than applications. While Carpenter and Grossberg have not explicitly developed a unified theory of cognition they have modeled a remarkable range of cognitive processes and has done so using an approach more sympathetic to a systems view of cognition than is typical in connectionist modelers. A good example of this approach can be found in their Adaptive Resonance Theory (ART) [Carpenter and Grossberg, 1987; Grossberg, 1987]. Superficially ART looks similar to many connectionist learning systems in that it is essentially a classification system, but it was developed to specifically address many of the shortcomings of such models. ART takes a feature vector as an input and uses it to provide a classification of the input. For example, a typical task would be to recognize hand-written numbers. The features would consist of the presence or absence of a pen stroke at different spatial locations.

One of the problems that ART was designed to address was what Grossberg [1987] referred to as the "stability-plasticity" dilemma. This is essentially a problem of how much new knowledge should impact what has been learned before. For example, a system that has been trained to recognize horses might have a problem when confronted with a zebra. The system could either change its representation of horses to include zebras, or it could create a separate representation for zebras. This is a significant issue for neural network models because they achieve a great deal of their power by having multiple representations share structure. Such sharing is useful for building compact representations and for automatic abstraction, but it also means that new knowledge tends to constantly overwrite what has come before. Among the problems this raises, is "catastrophic forgetting" as mentioned previously.

The stability-plasticity dilemma was addressed in part in ART through the introduction of a vigilance parameter that adaptively changed according to how well the system was performing. In some cases, for example, the system would be extremely vigilant and would require an unusually high degree of match before it would recognize an input as being familiar. In cases where inputs were not recognized as familiar, novel structure was created to form a new category or prototype. Such a new category would not

share internal structure directly with previously learned categories. Essentially when vigilence is high the system creates "exemplars" or very specialized categories, whereas when vigilence is low ART will create "prototypes" that generalize across many instances. This makes ART systems attractive since they do not commit fully either to exemplar or prototype models, but can exhibit properties of both, as seems to be the case with human categorization.

In ART systems an input vector activates a set of feature cells within an attentional system, essentially storing the vector in short-term memory. These in turn activate corresponding pathways in a bottom-up process. The weights in these pathways represent long-term memory traces and act to pass activity to individual categories. The degree of activation of a category represents an estimate that the input is an example of the category. In the meantime the categories send top down information back to the features as a kind of hypothesis test. The vigilence parameter defines the criteria for whether the match is good enough. When a match is established the bottom up and top down signals are locked into a "resonant" state, and this in turn triggers learning, is incorporated into conciousness, etc.

It is important to note that ART, unlike many connectionist learning systems, is unsupervised-it learns the categories without any teaching signals.

ART has since been extended to a number of times, to models including ART1, ART2, ART3, and ARTMAP. Grossberg has also tied it to his FACADE model in a system called ARTEX [Grossberg and Williamson, 1999]. These models vary in features and complexity, but share intrinsic theoretical properties. ART models are self-organizing (i.e., unsupervised, though ARTMAP systems can include supervised learning) and consist of an attentional and an orienting subsystem. A fundamental property of any ART system (and many other connectionist systems) is that perception is a competitive process. Different learned patterns generate expectations that essentially compete against each other. Meanwhile, the orienting system controls whether or not such expectations sufficiently match the input-in other words it acts as a novelty detector.

The ART family of models demonstrate many of the reasons why working with connectionist models can be so attractive. Among them:

- The neural computational medium is natural for many processes including perception. Fundamental ideas such as representations competing against each other (including inhibiting each other) are often difficult to capture in a symbolic model. In a system like ART, on the other hand, a systemic property like the level of activation of a unit can naturally fill many roles from the straightforward transmission of information to providing different measures of the goodness of fit of various representations to input data.
- The architecture of the brain is a source of both constraints and ideas. Parameters, such as ART's vigilance parameter, can be linked directly to real brain mechanisms such as the arousal system. In this way what is known about the arousal system provides clues as to the necessary effects of the mechanism in the model and provides insight into how the brain handles fundamental issues such as the plasticity-stability dilemma.

69.5 Summary

Unlike many disciplines in computer science there are no provably correct algorithms for building cognitive models. Progress in the field is made through a process of successive approximation. Models are continually proposed and rejected; and with each iteration of this process the hope is that the models come closer to a true approximation of the underlying cognitive structure of the brain. It should be clear from the preceding sections that there is no "right" way to do this.

Improvements in cognitive models come from several sources. In many cases improvements result from an increased understanding of some aspect of cognition. For example, neuroscientists are constantly getting new data on how neurons work, how they are connected, what parts of the brain process what types of information, etc. In the meantime models are implemented on computers and on robots. These implementations provide direct feedback about model quality and shortcomings. This feedback often will

lead to revisions in the models and sometimes may even drive further experimental work. Because of the complexity of cognition and the number of interactions amongst parts of the brain it is really the case that definitive answers can be found; which is not to say that cognitive scientists do not reach consensus on any issues. Over time, for example, evidence has accumulated that there are multiple memory systems operating at different time scales. While many models have been proposed to account for this there is general agreement on the kinds of behavior that those models need to be able to display. This represents real progress in the field because it eliminates whole classes of models that could not account for the different time scales. The constraints provided by data and by testing models work to continually narrow the field of prospective models.

Defining Terms

Back-propagation A method for training neural networks based upon gradient descent. An error signal is propagated backward from output layers toward the input layer through the network.

Cognitive band In Newell's hierarchy of cognition, the cognitive band is the level at which deliberate thought takes place.

Cognitive map A mental model. Often, but not exclusively, used for models of large-scale space.

Connectionist A term used to describe neural network models. The choice of the term is meant to indicate that the power of the models comes from the massive number of connections between units within the model.

Content addressable memory Memory that can be retrieved by descriptors. For example, people can remember a person when given a general description of the person.

Feed forward Neural networks are often constructed in a series of layers. In many models, information flows from an input layer toward an output layer in one direction. Models in which the information flows in both directions are called **recurrent**.

Graceful degradation The principle that small changes in the input to a model, or that result from damage to a model, should result in only small changes to the model's performance. For example, adding noise to a model's input should not break the model.

Necker cube A three-dimensional drawing of a cube drawn in such a way that either of the two main squares that comprise the drawing can be viewed as the face closest to the viewer.

UTC Unified Theory of Cognition.

References

Amit, D.J. (1995). The Hebbian paradigm reintegrated: local reverberations as internal representations. *Behavioral and Brain Sciences*, 18(4), 617–657.

Ballard, D.H. (1999) *An Introduction to Natural Computation*, Cambridge, MA: The MIT Press.

Booker, L.B. (1982). Intelligent Behavior as an Adaptation to the Task Environment. Ph.D. dissertation, The University of Michigan.

Carpenter, G.A. and Grossberg, S. (1987). A massively parallel architecture for a self-organizing neural pattern recognition machine. *Computer Vision, Graphics and Image Processing*, 37, 54–115.

Chown, E. (1999). Making predictions in an uncertain world: environmental structure and cognitive maps. *Adaptive Behavior*. 1–17.

Chown, E. (2002). Reminiscence and arousal: a connectionist model. *Proceedings of the Twenty Fourth Annual Meeting of the Cognitive Science Society*. 234–239

Chown, E., Jones, R.M., and Henninger, A.E. (2002). An architecture for emotional decision-making agents. In *The proceedings of Autonomous Agents and Multi-Agent Systems '02*.

Chown, E., Kaplan, S., and Kortenkamp, D. (1995) Prototypes, location, and associative networks (PLAN): towards a unified theory of cognitive mapping. *Cognitive Science*, 19, 1–51.

Clark, A. (2001). *Mindware: An Introduction to the Philosophy of Cognitive Science*. New York: Oxford University Press.

Craik, K.J.W. (1943) *The Nature of Exploration*. London: Cambridge University Press.

Dawkins, R. (1986). *The Blind Watchmaker*. New York: W.W. Norton & Company.
Dreyfus, H. (1972). *What Computers Can't Do*. New York: Harper & Row.
Hebb, D.O. (1949). *The Organization of Behavior*. New York: John Wiley.
Fodor, J.A. and Pylyshyn, Z.W. (1988). Connectionism and cognitive architecture: a critical analysis. *Cognition*, 28, 3–71
Grossberg, S. (1987). Competitive learning: from intereactive activation to adaptive resonance. *Cognitive Science*, 11.
Grossberg, S. and Williamson, J.R. (1999). A self-organizing neural system for learning to recognize textured scenes. *Vision Research*, 39, 1385–1406.
Jones, R.M., Laird, J.E., Nielsen, P.E., Coulter, K.J., Kenny, P.G., and Koss, F., (1999). Automated intelligent pilots for combat flight simulation. *AI Magazine*, 20(1), 27–41.
Kaplan, R. (1993). The role of nature in the context of the workplace. *Landscape and Urban Planning*, 26, 193–201.
Kaplan, R. and Kaplan, S. (1989). *The Experience of Nature: A Psychological Perspective*. New York: Cambridge University Press.
Kaplan, S. and Peterson, C. (1993). Health and environment: a psychological analysis. *Landscape and Urban Planning*, 26, 17–23.
Kaplan, S., Sonntag, M., and Chown, E. (1991). Tracing recurrent activity in cognitive elements (TRACE): a model of temporal dynamics in a cell assembly. *Connection Science*, 3, 179–206.
O'Keefe, M.J. and Nadel, L. (1978). *The Hippocampus as a Cognitive Map*. Oxford: Clarendon Press.
Kinsbourne, M. (1982). Hemispheric specialization and the growth of human understanding. *American Psychologist*, 34, 411–420.
Lachter, J. and Bever, T. (1988). The relationship between linguistic structure and associative theories of language learning — A constructive critique of some connectionist teaching models. *Cognition*, 28, 195–247.
Laird, J.E. (2003). The Soar 8 Tutorial. http://ai.eecs.umich.edu/soar/tutorial.html.
Laird, J.E., Newell, A., and Rosenbloom, P.S. (1987). Soar: an architecture for general intelligence. *Artificial Intelligence*, 33, 1–64.
Laird, J.E., Rosenbloom, P.S., and Newell, A. (1984). Towards chunking as a general learning mechanism. *Proceedings of the AAAI'84 National Conference on Artificial Intelligence*. American Association for Artificial Intelligence.
Lenat, D. and Feigenbaum, E. (1992). On the thresholds of knowledge. In D. Kirsh (Ed.), *Foundations of Artificial Intelligence*. MIT Press and Elsevier Science. 195–250.
McCloskey, M. (1991). Networks and theories: the place of connectionism in cognitive science. *Psychological Science*, 2(6), 387–395.
McCloskey, M. and Cohen, N.J. (1989). Catastrophic interference in connectionist networks: the sequential learning problem. In G.H. Bower, Ed. *The Psychology of Learning and Motivation* Vol. 24, New York: Academic Press.
Newell, A. (1990). *Unified Theories of Cognition*. Harvard University Press: Cambridge, MA.
Rochester, N., Holland, J.H, Haibt, and Duda, W.L. (1956). Tests on a cell assembly theory of the action of the brain, using a large digital computer. *IRE Transactions on Information Processing Theory*, IT-2.
Rumelhart, D.E. and McClelland, J.L., Eds. (1986). *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, The MIT Press: Cambridge, MA.
Squire, L.R. (1992). Memory and the hippocampus: a sythesis from findings with rats, monkeys, and humans. *Psychological Review*, 99, 195–231.
Tooby, J. and Cosmides, L. (1992). The psychological foundations of culture. In J. Barkow, L. Cosmides, and J. Tooby, Eds., *The Adapted Mind*, New York: Oxford University Press, 19–136.

Further Information

There are numerous journals and conferences on cognitive modeling. Probably the best place to start is with the annual conference of the Cognitive Science Society. This conference takes place in a different city

each summer. The Society also has an associated journal, *Cognitive Science*. Information on the journal and the conference can be found at the society's homepage at http://www.cognitivesciencesociety.org.

Because of the lag-time in publishing journals, conferences are often the best place to get the latest research. Among other conferences, Neural Information Processing Systems (NIPS) is one of the best for work specializing in neural modeling. The Simulation of Adaptive Behavior conference is excellent for adaptive systems. It has an associated journal as well, *Adaptive Behavior*.

A good place for anyone interested in cognitive modeling to start is Allen Newell's book, *Unified Theories of Cognition*. While a great deal of the book is devoted to Soar, the first several chapters lay out the challenges and issues facing any cognitive modeler. Another excellent starting point is Dana Ballard's 1999 book, *An Introduction to Natural Computation.* Ballard emphasizes neural models, and his book provides good covereage on most of the major models in use. Andy Clark's 2001 book, *Mindware: An Introduction to the Philosophy of Cognitive Science*, covers much of the same ground as this article, but in greater detail, especially with regard to the debate between connectionists and symbolists.