Probing and Prediction:
A Pragmatic View of Cognitive Modeling

Robert M. French
Service de Psychologie du Travail
Université de Liège
french@segi.ulg.ac.be

Axel Cleeremans
Laboratoire de Psychologie Industrielle
et Commerciale
Université Libre de Bruxelles
axcleer@ulb.ac.be

Abstract

This paper examines the role of computational modeling in psychology. Since any model of a real world phenomenon will fail at some level of description, we suggest that models can only be understood (and evaluated) with respect to a given level of description and a specific set of criteria associated with that level. We suggest a pragmatic view of the main advantages of instantiating psychological theories as computer simulations. We suggest that the main function of computational modeling is to support a process of "probing and prediction" by which models can be interacted with in a way that provides both guidance for empirical research as well as sufficient depth to support interactive modification of the underlying theory. Within this framework we briefly develop a way of comparing the quality of different models of the same phenomenon. We argue that models gain explanatory power as well as practical usefulness when they are emergent, that is, when they provide an account of how the principles of organization at a given level of description constrain and define structure at a higher level of description. For this reason, connectionist models would appear to provide the most fruitful modeling framework today.

Introduction

While the notion that computation lies at the root of cognition goes back at least to Hobbes's claim in the mid-17th century that "Reasoning is reckoning", computational modeling in general, and connectionism in particular, has a relatively short history. This short history has not prevented the computational approach from having a significant impact on psychological theorizing. One would indeed be hard pressed to find a cognitive psychologist who would deny that behavior is ultimately the result of some form of "computation." Curiously, however, while the use of computational metaphors to describe theories about cognition is widespread, relatively few psychologists actually use computer simulations in their research, and often resist doing so on principled grounds. For instance, some authors reject connectionist models because they feel that they are so complex as to be essentially intractable, and hence inadequate as theories of cognition (e.g., McCloskey, 1991). In this paper we hope, by way of explanation and example, to shed light on the value of computational modeling in psychology.

We will begin by considering a number of different models, ranging from string-and-paper-cup contraptions that simulate squawking chickens to low-level neuronal computer models of thalamocortical oscillations in which virtually all of the parameters are taken from

the empirical neurobiological literature. We will suggest, perhaps somewhat surprisingly, that all of these models have a level of description at which they are perfectly valid.

We then attempt to provide a principled basis for distinguishing the quality of various models of a given phenomenon. Most importantly, we suggest that:

- models must imperatively be associated with a given level of description and
- the main function of computational modeling is to support an interactive process of "probing and prediction."

A collection of models

We begin by several concrete examples of models (Cleeremans & French, 1996).

Walking down the streets of Prague, one hears chickens squawking. But when one looks for the chickens, one finds only a skilled human with a homemade device consisting of a little cup from which dangles a string covered with rosin. Pulling on the string produces a sound that resembles chicken-squawking so well that, for all intents and purposes, it is perfect. But what about the "model" used to produce the squawking? Does it tell us anything at all about how chickens actually achieve their squawks?

Consider now the following model of a professional basketball player. We know that this player has a field shooting average of exactly 0.500. So we take a coin and put it in a box. We shake the coin and whenever a head comes up, we will say that a basket has been made; a tail will mean a missed bucket. Now we add a little window-dressing to our device. After four straight heads, it will be rigged to say, "Hey, guys, I'm hot!". After six straight "baskets", it will say, "Hit me, guys, I can't miss!". After eight, it will say, at increased volume, "Feed me, feed me, the basket is bigger than a house!". If we are to believe a study by Gilovitch and Tversky (see Gilovitch, 1991) of the shooting records of the Boston Celtics over an entire season, the coin-in-a-box model will produce "shooting patterns" that are identical to those of a real player. In short, in our example, after a "streak" of five baskets (five heads) the chances of making the next basket (head) is, well, exactly 0.500. How good is our coin-in-the-box as a model of a professional basketball player with a 0.500 field goal shooting average?

What about "full-scale theme parks" that simulate cities? In Orlando, Florida, for example, a simulated version of Key West, Florida, is being built (Booth, 1996)? The real Key West is a small tropical island city at the end of a string of islands off the south tip of Florida. It has been home to pirates, sailors, smugglers, artists, writers (among them, Ernest Hemingway) and all manner of romantics for the past two hundred years. Now a full-scale simulation of the famous city is being created in Orlando. You will be able to walk down "Duval Street", get a feeling for the eclectic tropical atmosphere, explore shady back alleys, observe the lifestyles of (paid) local residents, and stroll on sandy beaches ("gently washed by mechanically created waves") simulating with great precision the beaches in the real Key West. (This last point is particularly amusing since the original Key West beaches were anything but sandy, consisting mostly of coarse, unpleasant "coral sand". The local chamber of commerce, realizing that these beaches did not fit with most tourists' notion of a sandy beach, hit upon the idea of "fixing" the natural beaches by trucking in many thousands of tons of fine white sand from the beaches of Miami to "simulate" a sandy beach. Thus the Orlando Key West's beaches are therefore really simulations of simulations.) Of course, as one of the architects of the simulation says, "a certain decadence" of the real Key West will be left out of the simulation. What does our understanding of the simulated Key West tell us about the workings of the real Key West?

Finally, consider *Deep Blue*, the chess-playing program created by IBM that recently beat the human world chess champion, Garry Kasparov, once and achieved a draw twice in a 6-game match played under international competition conditions. *Deep Blue*, like its illustrious predecessor, *Deep Thought*, plays chess by "look-ahead", in other words, by examining the consequences of a vast number of moves, possible counter-moves by the opponent, etc. In the case of *Deep Blue*, the program is capable of checking on the order of a quarter of a *billion* different board positions a second. It can look ahead at all possible board positions up through its next seven moves (14 counting its opponent's moves) before determining what piece to move. Does this extraordinary demonstration chess playing tell us anything about human chess playing? How does *Deep Blue* rate as a model of human chess playing?

Intuitively, what seems to be missing from all of these models is some account of the *mechanisms* underlying chicken squawking, basketball shooting, city dynamics or chess playing. One might argue that until these underlying mechanisms are also modeled we cannot say anything about the "real thing." While there is certainly some truth to this, one must be very careful.

Consider connectionist models. It is known, with something approaching certainty, that feedforward backpropagation does not exist in the brain, in spite of numerous unsuccessful efforts (Crick, 1989) designed to find something that could reasonably be said to "backpropagate" error signals to upstream neurons. As surely as we know that basketball players are not driven by the flipping of an internal coin, we know that real neural networks do not work with backpropagation. And yet no one thinks twice when these ubiquitous feedforward backpropagation networks are used to analyze high-level cognitive phenomena, such as speech production (e.g., Sejnowski & Rosenberg, 1987), sentence parsing (Elman, 1990) or word recognition (e.g., Seidenberg & McClelland, 1989); mid-level phenomena such as implicit learning (e.g., Cleeremans, 1993); or even low-level neural phenomena, such as dyslexia (e.g., Plaut & Shallice, 1993; Plaut, 1995). All of these models have been very successful in accounting for empirical data, yet all of them are based on the completely unsupported notion that learning takes places through back-propagation. In what ways are these models any different than *Deep Blue*'s highly successful performance at playing chess?

Finally, consider neuronal models (e.g., Miles, Traub & Wong, 1988; Golomb, Wang, & Rinzel, 1995) that hew to the constraints of experimental neurophysiology with unparalleled rigor. Ionic channels, sodium and calcium flows are all modeled to match experimental findings, connection schemes are copied from real neural patterns, neurotransmitter levels are carefully controlled and the Hodgkin-Huxley equations rigorously respected. From these models, neuronal spiking patterns can be produced and predicted. Higher lever oscillatory firing patterns among groups of "neurons" can be observed, predicted, and modified by changing any of a large number of experimentally observed parameters. Now, you say, *this* is real modeling. But this is only a "view from above" by cognitive modelers for whom synaptic modeling is at an almost unimaginably low level of detail and, therefore, must be closer to the hard reductionist truths of chemistry and physics. However, this feeling is illusory: These detailed synaptic models are routinely criticized by neurologists as oversimplifications of real neural events.

The central issue is whether neural models of either the connectionist variety or of the low-level synaptic type, are in some fundamental way different from the "obviously wrong" models of chicken-squawking and basketball-shooting? In this paper we will suggest, perhaps somewhat surprisingly, that all of the models described above are, in fact, perfectly appropriate at a particular level of description. Before we develop this argument, however, we need to root it in a discussion of the main issues involved in making inferences from data to theory. As we attempt to show in the next section, the relationship between data and theory is often considerably more complex than it may first appear.

Inference in psychology

Psychology in general is confronted with two problems. The first problem is to define and identify the measurable behaviors that correspond to a given phenomenological concept. The second problem is to identify the cognitive processes and representations that produce the observed behavior.

Much of the history of research in psychology can be understood as an attempt to establish relatively objective behavioral definitions for phenomenological constructs. Indeed, some contemporary areas of psychology (e.g., implicit learning) are replete with definitional issues. But even if we were able to agree on a set of measurable behaviors corresponding to a particular concept, we would still be confronted with the issue of understanding the causes of the observed behaviors. Making inferences from observed patterns of data to the underlying mechanisms is precisely what modeling and theory-building are all about.

A single observable behavior, however, may arise from a variety of causes. Our environment offers many examples that illustrate this simple point. For instance, one's television may fail to operate for a number of different reasons: It may be unplugged, the batteries in the remote control may be dead, a fuse may have blown, the cables supplying electricity to the tube may have become loose, and so on. At a given level of description, radically different (low-level) causes may result in the same (high-level) symptom. Any complex system that relies on many interacting components organized at different levels of description in order to produce some identifiable high-level behavior is likely to exhibit this many-to-one relationship between causes and effects. Many psychological constructs are obviously prone to the same kind of difficulty, even if a perfect definition of the phenomenon itself were possible.

Thus, while a skilled technician could certainly open up the television and precisely identify the cause of the failure, she can only do so because she has a detailed theory of how televisions work. Psychologists, unlike TV technicians, do not yet possess similarly detailed theories about the working of the mind. In order to develop such theories of mind, the architecture and mechanisms underlying cognition must be inferred from behavior. But a given behavior does not imply a unique cause. And there are now many examples of computational systems that are functionally (behaviorally) equivalent despite being based on radically different processing principles. For instance, many learning systems based on exemplars turn out to be able to produce abstract behavior and to behave in a rule-like manner without encoding rules explicitly. Likewise, the performance of symbolic computational systems based on chunking (Servan-Schreiber & Anderson, 1990; Laird, Rosenbloom, & Newell, 1985; Rosenbloom, Newell, & Laird, 1990) overlaps largely with the performance of the Simple Recurrent Network (Cleeremans & McClelland, 1991) in accounting for artificial grammar learning tasks performance (see Berry and Dienes, 1993; Cleeremans, 1993 for discussions). Some authors even go as far as claiming that many of these models are not empirically differentiable (Barsalou, 1990; Goldstone & Krushke, 1994). In contrast to these authors, we believe that it is possible to identify general methodological principles that can be used to preferentially select one model over another.

Overlapping models and levels of description

What are we to make of these overlapping models? Should some be taken as wrong and others as correct, even though they are all equally successful in accounting for performance? Or should they all be taken as correct at some level of description? Our view is that the latter position is the appropriate one. In other words, *the accuracy of models can only be judged with respect to a particular level of description and a specific set of criteria associated with that level* (see French, 1995, p. 146-48). In general, levels of description have a hierarchical

character. In other words, a given level of description will often emerge from a lower level of description. For example, a thunderstorm can be described by various high-level meteorological phenomena such as high winds, rain, lightning, which can, in turn, be described by the movement of air molecules, the creation of voltage differentials, etc. But this hierarchical emergence is not a necessary condition of modeling. Consider models of light. For some purposes, physicists still use Maxwell's equations, based on wave theory. For other purposes (like in a statistical model of scintillations on a TV screen) they may even use a simple particle model. And in still other applications they choose a quantum-mechanical description, which is neither a wave nor a particle model, but something different from both. None of these levels emerges hierarchically from the others as they did in the thunderstorm example. But the point remains the same: the level of description chosen is highly dependent on the pragmatic context in which the model is to be used.

To take a more psychologically oriented example, consider the effects of practice. In a wide variety of domains, performance improvement at a given task follows a power law: Changes in performance are large early in training, and tend to asymptote as training progresses, in a way that is best characterized by a power function. It is clear that the statement that the effects of practice follow a power law is merely descriptive: In and of itself, the equation does not do anything. Nevertheless, the equation is a model, in that it can imitate the data. It is merely a descriptive one, however, in that it fails to provide an account of the mechanisms that are involved.

Newell (1990; see also Laird, Rosenbloom & Newell, 1985) has proposed chunking as the mechanism responsible for the effects of practice. Performance improves because learning initially involves decomposing a task in many subtasks that need to be tackled separately — chunks. Through repeated exposure to the task, chunks can be combined, in a typically hierarchical way, to form more complex chunks. Each complex chunk can then be processed all at once without requiring further decomposition, and the resulting reduced computational overhead is sufficient to account for the speed-up in processing. This idea is so simple and powerful that Newell (1990) has proposed to incorporate chunking as the only learning mechanism in the production system architecture SOAR. SOAR is a powerful symbolic modeling environment that consists of a large collection of production rules that link conditions to actions, and which assumes that most of our cognitive activity can be described by processes of search in problem spaces. SOAR has been used to model performance in a wide variety of problem-solving and perceptual tasks. One of the central properties of SOAR is that through training, rules can be chunked together in the way described above so as to improve performance. Previously separate production rules are combined over training as the system detects systematic sequences of rule firings. Several rules can then become combined into a single rule that fires whenever the conditions of the first rule in the chain are fulfilled, and that outputs the actions of the last rule in the chain.

Chunking rules in this manner produces power-law improvements in performance, and SOAR is therefore an explanatory model with respect to the effects of training in that it does provide a well-defined mechanism to account for how practice at a task results in power-law like changes in performance. Does this mean, however, that we muist assume that people also acquire symbolic chunks when learning? Not necessarily. In several simulations, Elman (1990) demonstrated that the Simple Recurrent Network (henceforth, SRN) can learn to predict each successive element of sequences presented one element at a time. The SRN elegantly solves the problem of representing time by learning how to use a representation of the temporal context that the network develops itself over training through the back-propagation algorithm (see Cleeremans, Servan-Schreiber & McClelland, 1989). In one simulation, Elman generated a long random sequence of *b*'s, *d*'s and *g*'s and then replaced each letter by a group of letters according to the following rules: “b” was replaced by “ba”,

"d" was replaced by "dii", and "g" was replaced by "guuu". From the point of view of a system that is attempting to predict each element of the expanded sequence, the consonant elements (*b*, *d*, and *g*) are completely unpredictable since their original order was randomly chosen, but the vowel elements (*a*, *i*, and *u*) are fully predictable, because both their identity as well as their number are determined based on which consonant just occurred. After training on this material, the network learned this regularity perfectly, and produced low prediction errors for all the vowel elements, but high prediction errors for the consonant elements. So, for example, the network cannot predict when a *g* will occur, but once it has seen a *g*, it can predict almost without error the subsequent appearance of the three *u*'s. Thus, a plot of the error (the difference between the network's prediction and the actual successor) as it varies over successive elements shows the typical sawtooth pattern that would normally indicate that the system has chunked the sequence in small fragments corresponding to the *ba*, *dii*, and *guuu* sub-sequences in the input. But there is no clear evidence of any such chunks in the network's internal representations. The chunks exist at one level of description (the input sequence) and are a perfectly appropriate description at that level, but at the level of description of hidden-layer representations within the network they anything but chunk-like; rather, they are graded and distributed.

Hence, at a high-level of description, SOAR's account of the empirical data is perfectly correct: Chunking does indeed appear to be the crucial mechanism in understanding practice effects. Yet, at another level of description, it is incorrect to assume that the chunks need to be explicitly represented as chunks within the system. Instead, as Elman's work shows, the chunks may be purely functional and not be represented as such anywhere. Should we then reject SOAR based on the fact that chunks may be purely functional? Certainly not. SOAR's account is very valuable at the level of description at which SOAR operates, that is, at the symbolic level.

This does not mean, however, that our basic assumptions about how the cognitive system works are completely neutral with respect to how we think about the relationships between phenomenology, behavior, and cognition. We believe that such assumptions can have considerable impact on the conduct of research. For instance, Cleeremans (in press) argues that one of the central assumptions of the symbolic framework is that the relationship between observable patterns of behavior and the internal representations and processes that produce them is direct and transparent (see also Clark, 1990). The tacit adoption of this assumption has often resulted in unwarranted inferences from data to the cognitive architecture. A classic case of this is the conclusion that because people behave as though they possess rules, that there is probably an explicit representation of these rules somewhere in their memory. Another example and one which will be considered in more detail in this paper is the inference that double dissociations necessarily entail the existence of separable underlying modules.

An important function of computational models in psychology is therefore to provide us with metaphors with which to think about cognition. D. O. Hebb used to emphasize that a theory did not need to be right in order to be informative and to guide researchers in the right direction (Harnad, 1985). Like Hebb, we wish to emphasize that from our perspective, models — even when they are "obviously wrong" at some level of description — can still potentially provide insights into the space of theoretical accounts of a given phenomenon. Rather than attempting to provide a complete (and, consequently, impossible) description of reality, the main issue is the degree to which a given model is useful in helping us ask new questions about reality. In the next section, we argue that one of the main determinants of whether a model will be useful in this specific way is whether or not the model's primitives are cast at a level of description that is lower than the level of description that is appropriate to characterize the phenomena the model is meant to explain.

Levels of description and explanatory power

The objects of psychological inquiry are complex systems that afford analysis at different levels of description (Marr, 1982). Since the different levels all coexist, one might ask whether the level of description at which theories are framed really matters. We will argue that what is crucial is the potential ability to explain one level in terms of the primitives of another. In other words, our understanding of a given phenomenon gains explanatory power only when we can provide a causal account of a particular phenomenon in terms of the entities and organizing principles at a lower level of description than the phenomenon itself.

The main argument of this section, therefore, is that what makes a model useful and interesting is the fact that its natural primitives are cast at a level of description that is lower than the level at which the observable regularities that the model are meant to explain are described. According to this analysis, models can be classified into different categories according to their relationship with the phenomena they are meant to capture.

Models can, for example, be purely descriptive, meaning that they can provide adequate descriptions of the data but provide no explanatory mechanisms as to why the data appears as it does. For instance, a mathematical formula may accurately describe the parabolic arc of a cannonball. While this is a clear improvement over pre-Gallilean depictions of cannonball flight (i.e., a straight-line ascent, followed an arc of a circle, concluding with a straight-line descent), it still provides no account of *why* the arc should be a parabola as opposed to the earlier incorrect description. A theory of the underlying causes was needed for that, a theory later provided by Newton. Likewise in psychology, statistical theories of learning (e.g., Estes, 1957) relied heavily on mathematical descriptions of the changes in performance caused by learning, but seldom included an account of underlying mechanisms.

Explanatory models, by contrast, do attempt to bridge the gap between observable regularities and the representations and processes that might be responsible for these regularities. Clark (1990) distinguishes between "semantically transparent" explanatory models that use as their representational and processing medium conceptual elements from same semantic level as the phenomena they are attempting to explain and emergent explanatory models that include an account of how lower-level mechanisms might give rise to the higher-level observable phenomena. Most symbolic models fall into the first category, in that they appeal to principles that are best thought of as having a direct and transparent relationship with the data they are meant to explain. SOAR's account of learning, for instance, is entirely built around mechanisms that implement chunking. There is, therefore, in SOAR, a direct conceptual relationship between the phenomenon to be explained and the mechanisms used to explain the phenomenon.

Connectionist models, by contrast, are examples of the latter type of model, in that they are emergent: The principles that govern representation and processing in connectionist networks are cast at a lower level of description than the level of description that is appropriate to describe their behavior, and bear no transparent relationship with the phenomena that they are able to account for. For instance, Elman's SRN can produce chunking-like performance without explicitly resorting to representing chunks. Instead, the chunking performance emerges out of the interaction between the constraints that characterize the task and the principles that characterize the network's architecture and processing. This sub-symbolic account of cognition (Smolensky, 1988) constitutes in our view the main appeal of the connectionist framework, not only on philosophical grounds, but also for practical reasons. Indeed, one of the main arguments for the use of computational modeling in psychological research is that it helps formulate better theories because it enables them to be developed along the lines of an interactive "probing, prediction and modification" methodology that will be presented later in this paper.. Such interactive

development is much richer when the model's representational and processing medium is rooted in principles that apply to a level of description that is lower than the phenomena to be explained.

Of course, all emergent explanatory models are not equally valid. On what basis can we say that some emergent explanations of high-level phenomena are better than others? A detailed treatment of this question is beyond the scope of this paper, but the essence of the answer undoubtedly lies in the number and nature of the free parameters required by the causal level and the ability to continue probing-and-predicting at the newly introduced level of description. For a detailed examination of this question, see Forster (1988).

Theory-building with computational models

Building theories is of course what science is all about. In this section we would like to illustrate several different ways in which computational models help us formulate theories. A simple way to start this discussion consists of asking which criteria should be used to assess computational models. McCloskey (1991) identified three properties of theories by which their quality should be assessed: (1) generality, that is, whether the theory unifies existing data in a way that allows generalization, (2) testability, or whether the theory has identifiable components to which credit or blame in accounting for data can be assigned, and (3) specificity, that is, whether the theory is detailed enough that it can be contrasted with other competing theories in specific ways. To this list, Seidenberg (1993) added a fourth element: Explanatory power, or whether the theory appeals to independently motivated principles.

Computational models, according to McCloskey, are best thought of as instantiated (verbal) theories. Seidenberg (1993), on the other hand, stresses that theories should be explanatory, that is, that they should appeal to independently motivated principles. In contrast, our perspective on modeling is more pragmatic: What makes a theory useful and interesting, we contend, is the degree to which it can support a process that we call "probing and prediction", that is, whether it can be interacted with in a way that provides both guidance for empirical research as well as sufficient depth to support interactive modification. We illustrate this perspective by introducing five functions that computational models can serve in the context of developing theories. First, models can serve as simple existence proofs. Second, models can be demonstrations of new capabilities (e.g., spontaneous generalization). Third, models can be used to unify an existing body of empirical and theoretical research. Fourth, models form the basis of a "probing-and-prediction" research strategy. Finally, models are used for further theory development through interactive modification of the model in light of the results of probing and prediction. We now turn to a discussion of each of these functions.

Counterexamples, demonstrations, and existence proofs

Models can simply be vehicles to instantiate counterexamples, demonstrations and existence proofs (McCloskey, 1991). Assume there is some phenomenon P for which we hope to gain a better understanding. Further, assume that it is generally believed that "No system with the set of features S can produce P." If we can successfully build a system having the set of features S that can, in fact, produce P, then we have not only produced a counterexample, but we have also acquired a deeper understanding of P, by the very fact of developing a system that could produce it. Existence proofs fall into this category. Three examples from connectionist modeling will serve to illustrate this point.

One of the strong claims of traditional artificial intelligence (AI) has always been that transformational grammars can only be modeled in rule-based symbolic systems (Fodor & Pylyshyn, 1988). Chalmers (1990), however, showed how a RAAM connectionist architecture (Pollack, 1989) could also produce active-to-passive transformations. RAAM

networks are a variety of recurrent connectionist networks that are capable of learning how to process hierarchical structures. Because RAAM networks are instances of connectionist networks, and because connectionist networks rely on distributed representations rather than on high-level symbolic rules to process information, Chalmers' demonstration using the RAAM model constitutes an existence proof. Whether or not humans perform active-to-passive transformations by means of a RAAM-like architecture (as opposed to applying high-level grammar rules) is beside the point. Chalmers' demonstration simply says "There exists at least one way of performing active-to-passive transformations without high-level rules." It does not say, "This is how humans actually do active-to-passive transformations." Rather, this type of model can act as a bellwether for future research: If it turns out that the RAAM architecture, which does in fact contain some features that are relevant to real brains, can also be used to account for other data, then the case for language processing using connectionist networks grows stronger.

Another example is provided connectionist networks that can often be described as obeying rules without possessing anything like high-level rule representations. A well-known example is Rumelhart & McClelland's (1986) model of the acquisition of the past tense morphology. In their model, not only are regular verbs processed in the same way as exceptions, but neither are learned through anything like processes of rule acquisition. Similarly, humans exhibiting rule-like behavior cannot necessarily be assumed to possess high-level, explicit representation of the rules they seem to be following. The point is that observing sensitivity to some high-level regularity does not necessarily imply that the regularity itself is represented within the system as an explicit object of representation.

A final example of how models can produce complex emergent effects at the level of their observable behavior is provided by the work of Plaut and Shallice (1993) on deep dyslexia. Standard neuropsychological interpretations of clinical data rely heavily on an assumption that Farah (1994) describes as the "locality assumption", and which basically states that the cognitive system consists of a collection of functionally specialized processing modules that are structurally independent from each other. Double dissociations then receive seemingly natural interpretations: Damage to one specific module of the system results in deteriorated performance on tasks involving the function supported by the damaged module but has no effect on performance involving functions supported by other modules. Plaut (1995; see also Farah, 1994; Bullinaria & Chater, 1995) however, proposed a radically different interpretation of double dissociations by showing how a connectionist network can exhibit functional double dissociation despite not being organized in architecturally distinct processing modules.

Plaut and Shallice (1993) systematically damaged a connectionist network designed to produce phonological representations of concrete or abstract words when presented with their orthographic representation by randomly selecting and removing some connections from different processing pathways in the network. In so doing, Plaut and Shallice were able to have the network reproduce the double dissociation pattern observed with human patients. A detailed account of how the network was lesioned so as to reproduce observed patterns of dissociation is beyond the scope of this paper, but the point is simply that all pathways are equally involved in processing both concrete and abstract words. Contrary to most standard interpretations of double dissociations, this work demonstrates that double dissociations may not necessarily be attributable to explicit architectural modularization, but may instead be a consequence of functional specialization in the representational system of the network. In this way, the modeling work of Plaut complements the work of Dunn and Kirsner (1988), who arrived at essentially the same conclusions (i.e., a rejection of the standard modular interpretation of double dissociations) on purely logical grounds.

These findings have important implications for the present paper because they show that it is unwarranted to assume the existence of a simple relationship between observed data and underlying mechanisms. Observing a double dissociation does not necessarily entail the existence of separable underlying modules, just as observing rule-like behavior does not necessarily entail that the rules are represented as such in the system, and so on. In all these cases, simulation models were instrumental in demonstrating the plausibility of these new interpretations of high-level behavior.

Demonstrating new capabilities and improving implementations of old ones

Some models do certain things better than other models. Some do things more "naturally" than others. Indeed, some models do things that other models cannot do at all. One of the purposes of modeling is to explore new capabilities or to show how old capabilities can be achieved in novel ways. Although Samuel (1959) was not concerned with psychological plausibility, his checker-playing model showed that it was possible for a computer using "look-ahead-and-evaluate" and "weight-adjustment" techniques to actually learn to perform a high-level cognitive function.

After Samuel had demonstrated that a computer could be programmed to perform a particular high-level cognitive task (in this case, checker-playing), Newell et al. (see Newell & Simon, 1972; Newell, Shaw & Simon, 1957) went on to expand this work into a general cognitive architecture based on the notion that all of cognition consisted, in one way or another, of problem-solving. Their problem-solving technique, "means-ends analysis," was directly inspired from detailed protocols of humans solving problems. Computational modeling of high-level psychological processes was off and running. Whenever a roadblock was encountered — and there were many — new models were developed using new techniques to overcome the problem. Connectionist modeling was no different. By the beginning of the eighties, connectionist models were providing new means of understanding important aspects of human performance, such as spontaneous generalization, content-addressable memory, or graceful degradation in the presence of noise or damage (see McClelland, Rumelhart & Hinton, 1986). While some of these capabilities could be produced by traditional, symbolic models, in connectionist models they were a natural by-product of the networks' underlying principles of processing and representation.

Hence models not only provide new capabilities, as in the case of Samuel or Newell, Simon, & Shaw, but they also can provide new (and better, or more natural) ways of doing old things. An example from connectionism is the development of the SRN (Elman, 1990). By their very nature, feedforward backpropagation networks (BPN) are the quintessential stimulus-response engines. These pattern associators learn functions; in other words, no input can be associated with more than one output. Consequently a BPN could learn a sequence "A-B-C-D-E-F" by associating A with B, B with C, C with D, and so on. Then by starting with "A" and making each successive output the succeeding input, it could reproduce the originally learned sequence. But what about the sequence: "A-B-C-D-B-F"? The first "B" in the sequence is followed by a "C" while the second "B" is followed by an "F". In other words, a single input produces in one case a "C" and later an "F". The earliest attempts to solve this problem used a "sliding window" technique (e.g., Sejnowski & Rosenberg, 1987). Instead of having a single letter on input, the input consisted of an adjacent pair of letters, thus: "AB" mapped to "C", "BC" to "D", "CD" to "B", and so on. The problematic 1-to-2 mapping was eliminated and the second sequence could be learned. But what about sequences that require larger sliding windows, like "A-B-C-D-E-F-B-C-H"? Now, a window of size three is required for a network to learn the sequence. Problems were solvable with this technique, albeit in a rather ad hoc manner. Elman (1990) developed a network with a simple recurrent topology that could solve these sequence problems and did not rely on determining

the size of a sliding window for a particular problem. It relied instead on a small one-step internal memory that was fed back into the network at each time step. The Elman technique was not, strictly speaking, a new capability, but rather a better, more natural way of solving the problem of sequence learning than the cumbersome "sliding window" method.

Hence a significant function of computational modeling is to develop models which, in one way or another, demonstrate new capabilities. Subsequent "probing-and-prediction" will determine whether the ways of achieving these new techniques will be incorporated in later generations of models.

Unification

A third function of computational modeling is to reconcile previously disparate and possibly contradictory empirical findings or theoretical accounts. Mendeleev's development of the periodic table of the elements is probably one of the most outstanding examples of this function of modeling. He arranged the elements according to their atomic weight and, according to his model, was able to predict the characteristics of elements as yet undiscovered. Not surprisingly, it turned out that this model was inaccurate in a significant way — namely, that the proper arrangement is one based on atomic number, not weight — but, in a sense, this is a detail. His model unified a large body of disparate information. Once he had presented his model of unification, others were able to probe it and to bring it into better alignment with experimental evidence.

Perhaps the most widely known attempt at unification in psychology is provided by the work of Newell. In his book "Unified theories of cognition", Newell (1990) proposes that psychology attempt to develop wide-ranging theories of cognition and offers the symbolic model SOAR as the leading candidate for such an attempt. According to its authors, SOAR can potentially account for all of cognition. This claim, while it may prove incorrect, is certainly one to which models of cognition should aspire. Perhaps one of the most cogent criticisms of current connectionist modeling is that they tend to be very "problem-specific." There exist myriad connectionist architectures, with an extremely wide range of learning rules, connection topologies, activation rules, number of nodes, and so on. Often, the best reason that the authors of a particular model can come up with for why they used it is simply, "Because it works."

It is true that the basic principles of all connectionist models were spelled out by Feldman & Ballard (1982), but these are so broad that they apply equally well to localist networks, distributed networks, Hopfield networks, feedforward backpropagation networks, and Hebbian networks, to name just a few. And, while it may be too much to suppose that any single, undifferentiated connectionist architecture could produce all of cognition — even in the ultimate connectionist engine, the brain, there are many types of different neurons, connection schemes, firing capabilities corresponding to different areas of the brain — what is needed is a connectionist architecture that will be to connectionism, at least in part, what SOAR claims to be for traditional AI (Weaver, 1993). Nevertheless, a number of connectionist models that have succeeded in integrating large bodies of empirical research, and if no specific model can yet claim to be a "unified theory of cognition", the principles upon which most connectionist models are based (e.g., parallel processing, distributed representations, etc.) certainly have universal appeal.

Probing and prediction

How does one work with computational models? In this section, we attempt to answer this question by discussing a fourth function of computational models: their ability to support an interactive process of probing and prediction (Cleeremans & French, 1996). By this we mean that a model must be able to be probed and must be able to make predictions about the

phenomenon that it is modeling, rather than simply being able to reproduce the phenomenon being modeled. These are arguably the two most important points in understanding the endeavor of modeling.

By probing a model (French, 1995, p. 146; Hofstadter et al, 1995), we mean, in a figurative way, the ability to "ask the model questions", to study the responses that it gives, and to compare these responses with the responses to the same questions as obtained through empirical research on the phenomenon being modeled. Thus, the quality of the model must be assessed with respect to the particular probes that it is tested with. This is crucial. Too often the supporters and critics of a particular model do not clearly specify the criteria (i.e., the probes) with which they intend to test the model. This often results in wholesale rejection of models based on the fact, for instance, that they are unable to carry out a particular task, or that some detail of their mechanisms is inconsistent with empirical evidence.

The pragmatic perspective on modeling advocated in this paper makes it possible to distinguish between various types of models, and provides a method — probing and prediction — by which the quality of models can be judged. It also allows us to see how models can be modified to eliminate the discrepancies with reality exposed by probing and prediction.

Let us consider the example of chicken-squawking. If our probe is, "Can the squawk-producing agent [i.e., the model] fly?" then, of course, the answer is no. And, on the basis of this probe, the cup-and-string squawker is a very bad model of a "real" squawker (i.e., a chicken). However, if we focus, say, on the object from which the squawking emanates, then the model might turn out to be quite good and productive. For example, we might observe that in the model the squawking sound is caused by a vibrating string. From this observation we may conclude that perhaps chickens have a similar noise-making device somewhere and set about to find it. And we would discover that they do indeed have just such a device — vocal chords that are vibrated, not by being rubbed with a rosin-coated finger, but by air forced through them. And we examine how the noise is made in the model, by amplifying it with a tightly drawn membrane that forms the bottom of an amplification cup. This means that it might be reasonable to see if there is something similar in the chicken. In the chicken's case, its throat and mouth have similar megaphone designs. We could probe the model by, say, increasing or decreasing the length of the string. This would allow us to make predictions, with respect to the criterion of string length, about the changes in real chicken squawking that would be induced by modifying the length of its vocal chords. And those predictions could be checked. Similarly, say, for the size of the sounding box, or the hardness of the paper, the quantity of rosin on the string, and so on.

Probing and prediction, of course, go hand in hand. Each time that a particular aspect of the probing is changed (i.e., a different question is asked of the model), a prediction can be made about how the equivalent change would affect the real phenomenon. Discrepancies between predictions and actual performance provide impetus to modify the model, which can then be further refined.

In a similar vein we could test the model of Key West. If the probes involve the locations of particular streets, the time it takes to walk from the beach to Duval Street, the size of Sunset Pier, or even the feeling of Key West street life (the promoters promise a "casual and eclectic neighborhood [populated by paid] colorful characters"), then the Orlando Key West, with respect to these probes, will certainly be a good model. If, on the other hand, you want to get a feeling for the seamier side of Key West life, the sailor hangouts, the red-light district, the unemployment office for out-of-work shrimp fishermen, the Orlando model will be a very poor one.

Notice that the ability to carry out this type of probing means that models must have the flexibility to allow testing. The key to this type of probe-and-predict methodology that we

are suggesting is to be able to make many small variations on the model to observe their effects and to compare those effects with the real world situation (French, 1995, pp. 146–148). This then provides one of the prime ways of evaluating the quality of a particular model: How well does it support this process of probing and prediction?

This probe-and-predict evaluation strategy will allow us to view modeling of chess playing in a somewhat unexpected light.

Probing and prediction and the issue of computer chess

In the early days of artificial intelligence, chess playing was considered to be the quintessential example of a high-level human cognitive activity that might be successfully simulated by computers. From the late fifties through the mid-seventies, researchers attempted to take advantage of computers' extraordinary speed in their attempts to produce a world-class chess-playing program. But the goal turned out to be far harder than anyone had originally imagined. Curiously, the great difficulty of writing a world-champion chess program had one unsuspected benefit: It provided a striking example of a high-level cognitive activity that humans, with their comparatively slow and faulty neural circuitry, could do far better than machines. It showed us just how good our own neural computation algorithms had to be.

Consequently, by the mid-seventies work had begun to focus on "cognitive" heuristic-driven programs that attempted to incorporate high-level "cognitive" strategies used by grandmasters to play chess. These programs met with considerable success compared to the earlier brute-force programs, but they were still very poor by human standards. By the end of the 1970's a human chess player of average skill could still trounce any computer chess program in the world. But it was, nonetheless, widely believed that the "cognitive" approach to modeling computer chess playing would ultimately prove to be successful. Some authors even went so far as to predict that "There may be programs which can beat anyone at chess, but they will not be exclusively chess players. They will be programs of *general* intelligence, and they will be just as temperamental as people" (Hofstadter, 1979).

But by the mid-1980's the wind had again shifted with the advent of parallel processing and far faster computing hardware. The idea of endowing programs with high-level "conceptual" heuristics gave way to brute-force lookahead-and-evaluate techniques. By 1988, *Deep Thought*, was capable of evaluating 750,000 board positions a second and went on to beat a number of grand-masters. Eight years later, a successor to this program, *Deep Blue*, using the same type of brute-force lookahead strategy but with the capability of analyzing a quarter of a billion board positions a second, triumphed over the world chess champion, Garry Kasparov, even if the program ultimately lost the 6-game tournament by 4 games to 2. Chess experts currently give *Deep Blue* a rating of 2650, making it the 20th best player in the world.

In cognitive modeling circles, *Deep Thought* and *Deep Blue* are frequently used as examples of an obviously "non-cognitive" computer program that is able to (brilliantly) perform the high-level cognitive task of chess-playing. The program's non-cognitive nature is supposedly the result of its architecture that relies solely on looking ahead at many billions of board positions before choosing a move, something that we know the human brain does not do. They therefore conclude, wrongly in our opinion, that *Deep Blue* is not a cognitive model.

Those who maintain that *Deep Blue* is not a cognitive model have fallen into the level-of-description trap described earlier. *At the level of the cognitive activity of chess-playing, Deep Blue* certainly *is* a perfectly appropriate model. It is indistinguishable from a very good human chess player in its ability to move pieces around a chess board. Only once we have begun to *probe* the program, as Kasparov did in learning how to play against it (defeating it

handily in games 5 and 6 after losing the first game and drawing games 3 and 4) do subtle differences between *Deep Blue* and human chess players begin to appear. We cannot say: *Deep Blue*'s underlying mechanisms are known to be different than our own, therefore it is not a cognitive model. If we allow this reasoning, then *nothing* (except possibly another human being) would ever count as a cognitive model, because at some level the underlying substrate of the model will inevitably differ from the "real" carbon-based neural substrate of a human. Connectionist models of all kinds would be subject to this criticism. We suggest that the appropriate strategy is to start at the level at which the model does appropriately model the phenomenon under consideration (in this case, the level of actual chess-playing) and then begin probing "downward." The quality of the model will be determined by two factors:

- how deeply the model can be probed before differences compared to human players come to light (French, 1990);
- how many free parameters need to be added to bring the model once more into alignment with reality as revealed by probing at the new level of examination (Forster, 1989).

It is interesting to read Kasparov's description of his interaction with *Deep Blue*. He talks about the program's "understanding of certain positions" and of its having "its own psychology." When interacting with *Deep Blue* — by competing against it — it makes perfect sense to apply high-level chess concepts to describe the program's play. We can talk about *Deep Blue*'s "stalking a piece", "attempting to capture the center", "setting a trap", and so on, regardless of the brute-force substrate driving the program.

Probing and prediction and human cognition

Now let us extend our comments about *Deep Blue* to human cognition. Let us assume that a brute-force program, *Deep Hugh,* was developed that was able to successfully play, not chess, but the "human cognition game." In other words, it could pass a full-blown, hard-as-they-come Turing Test. French (1990) has discussed the extraordinary difficulty that any program that had not lived life like a human had would have in passing this test. In other words, were such a thing possible, careful probing would reveal no fundamental differences between *Deep Hugh*'s and the participating human's subcognitive underpinnings. Having passed a Turing Test that presumably would have included the type of subcognitive probing discussed in French (1990), there would be no way for us to detect that *Deep Hugh*'s concepts were any different than our own. Probing of *Deep Hugh* would reveal that its concepts would exhibit the same rich, deeply interwoven structure as human concepts and this, regardless of the substrate that produced them. Whether the program was ultimately driven by a "brute-force" substrate or a "pseudo-neural" substrate would be of no importance.

Interactive modification

The probing-and-prediction strategy leads naturally to the notion of interactive modification. The idea is that whenever a model does not fit the predictions produced by probing it, it should be modified. The modifications of the model would then gradually bring it in line with predictions. This opens up an entirely new set of probes to be applied to the modified model. Questions that could not be asked of the original model can now be asked of the new model. In this way, we progressively increase the depth of our model.

In certain cases the modifications will produce a new model that further probing will reveal to be seriously flawed, perhaps irreparably flawed. The classic case of this catastrophic break-down is perhaps the Ptolemaic model of the cosmos (see Burtt, 1932). The work of Galileo and Newton subsequently showed that the alternative model proposed by Copernicus

could accommodate much deeper probing without insurmountable problems (i.e., ultimately requiring the addition of fewer adjustable parameters than the Ptolemaic model), and further, that continued refinements could be made in the latter model.

This cycle of probing, prediction, comparison with reality, and model modification is what modeling is all about. This is why the question of levels of description is so important. Indeed, if the substrate on which the model is based is sufficiently rich, then probing-and-prediction will allow the model to be expanded downward towards ever more fundamental principles. If the substrate is poorly conceived, or even absent, then this type of probing will most likely dead end.

A number of examples serve to illustrate this point in the area of connectionist modeling. Our earliest example dates from the first computer model of Hebbian learning (Rochester, Holland, Haibt, & Duda, 1956). This research attempted to implement Hebbian learning as described in Hebb's "Organization of Behavior" (Hebb, 1949). In his book, Hebb carefully avoided any reference to inhibition, since at that time inhibitory effects had not been observed in real neuronal circuitry. When Rochester et al. (1956) built their model, however, they found that without inhibition, activation invariably spread everywhere. In other words, they had an extremely hard time controlling the spreading of activation in the model. When inhibitory connections were added, however, the problem was brought under control. This suggested the importance of inhibition in such networks, and inhibitory connections were indeed discovered in real neurons at about the same time.

Another example comes from current synaptic modeling. Certain researchers are attempting to build models of interconnected neurons that rigorously respect a wide range of neurophysiological data. One particular type of model (Thomas & Wyatt, 1995) models the interactions of thalamic neurons. Out of the lower level neuronal constructs and equations used in this model emerge the 3 Hz brainwave oscillations that seem to characterize epileptic patients during seizures. The idea is to use this model to explore what low-level changes will disrupt this oscillatory neural firing. The model can therefore be probed with respect to this criterion ("What can prevent these oscillations from occurring?"). Once this is known in the model, the corresponding question can be asked of neurophysiologists. (For instance, "De-inactivating T-channels prevents oscillation in our model, are there any ways to achieve this pharmacologically in real neurons?") The obvious advantage of this interactive method is that it is much easier to explore the space of possible solutions in a computer model than in real humans.

Finally, a somewhat more detailed example illustrates how modeling work was actually instrumental in understanding the data. Cleeremans & McClelland (1991) explored performance in a reaction time situation characterized by the fact that the locations at which successive events appeared were determined based on the generation rules specified by a probabilistic finite-state automaton. Thus, on each trial, the stimulus could appear at any screen location, but some locations were more likely than others depending on the previous sequence of stimuli. By comparing reaction times on predictable and unpredictable trials, Cleeremans and McClelland (along with many others, e.g., Nissen & Bullemer, 1987) showed that in this kind of situation, participants exhibit detailed sensitivity to the sequential constraints embedded in the stimulus material. Participants tended to be much slower in responding to stimuli that were inconsistent with the previous sequence (i.e., stimuli that had a low conditional probability of appearance at that particular moment) than to stimuli that were consistent. They proposed a connectionist model of performance for this task that instantiated the theory in the form of an SRN network. One crucial test of the adequacy of the theory consisted of showing that the response distributions of the model (i.e., activation strength) and of human participants (i.e., reaction times) both tended to approximate the distribution of the conditional probabilities of the occurrence of each stimulus in different

contexts. It turned out, however, that even though the model's responses correlated extremely well with the appropriate theoretical conditional probabilities, they did not correspond at all to human participants' responses. This discrepancy prompted a second, closer look at the human data. It turned out that a crucial factor that affects reaction time but not the model's responses had been overlooked. When simple mechanisms were added to represent the overlooked effects, the SRN model was then able to account for almost 90% of the variance associated with human responses — an extremely good fit of theory to data.. The point is that this interaction between theory and data analysis would most likely not have occurred without the simulation model.

Conclusion

In this paper, we have presented an essentially pragmatic view of computational modeling. We have suggested that

- the main function of computational modeling is to support an interactive process of "probing and prediction" through which the model can be interacted with in a way that provides both guidance for empirical research as well as sufficient depth to support interactive modification of the underlying theory;
- the quality of models can, and should, be judged on the basis of their ability to support this probing-and-prediction methodology;
- models, just as the systems they are models of, can only be understood with respect to a given level of description and a specific set of criteria associated with that level (and hence that even demonstrably wrong models can be useful);
- in general, models constructed from elements whose level of description is lower than the level of description of the regularities that the model is designed to account for are more likely to be able to support a probing-and-predicting analysis. From this perspective, connectionist models, because they are emergent, appear to offer the most interesting and productive avenue of research.

We would like to end this paper with a reflection on some of the potential problems involved in working with models. Standard, descriptive models are sometimes rejected because they seem to offer no more than the verbal descriptions of the phenomenon being modeled. Connectionist models, on the other hand, have been criticized (McCloskey, 1991) because they seem to offer a picture of cognition as essentially intractable. McCloskey argues that the inherent (and enormous) complexity of large connectionist networks is a major problem because if we do not understand the model any better than the observable data, then of what value is the theory instantiated by the model? If we accept McCloskey's criticism, does this mean we should refrain from simulating data until we clearly understand all of the aspects of what the model is doing? We think not. We view the process of developing theories as a process that ultimately involves the interactive exploration of both the empirical and modeling spaces. The very aspects of connectionist models that McCloskey criticizes are what make them attractive as a modeling tools — namely, their emergent, complex and dynamical behavior (van Gelder, 1995). Insofar as human-like behavior can emerge from these models, their underlying complexity may be rich enough to allow the interactive probing-and-prediction strategies that we have advocated above. Thus, McCloskey's characterization of connectionist models as animal models is quite appropriate, in that the challenge is to understand both the model and the data the model is meant to explain. We believe that the benefits of this kind of dual exploration far outweigh its potential drawbacks.

From this perspective, then, computational modeling then is not just another tool in the cognitive psychologist's toolkit. It is instead the most important tool of all. We maintain that to be truly useful, computational models should be rooted in principles that are specified at a lower level of description than the data they are meant to explain. The challenge is to understand how organization at some level of description produces effects at some higher level of description (e.g., "How does the brain produce mind?", "How can one produce symbol manipulation based on distributed patterns of activity in neural circuits?", "How can sequentially organized behavior arise from parallel processing mechanisms?", and so on).

In the final analysis, computational models of cognition are metaphors for the processes of human thought. As Dennett (1991, p. 455) has so elegantly put it: "It's just a war of metaphors, you say — but metaphors are not "just" metaphors; metaphors are the tools of thought.". Likewise, we believe computational models to be the essential tools of theory-building.

Acknowledgments

Robert French is supported by grant No. D.4516.93 from the Belgian National Fund for Scientific Research. Axel Cleeremans is a Research Associate of the Belgian National Fund for Scientific Research. We also wish to thank Luis Jiménez, Alain Content and Malcolm Forster for their many helpful comments on early drafts of this paper.

References

Barsalou, L.W. (1990). On the indistinguishability of exemplar memory and abstraction in category representation. In T.K. Srull & R.S. Wyer (Eds.), *Advances of social cognition (vol. 3): Content and process specificity in the effects of prior experiences.*

Berry, D.C., and Dienes, Z. (1993). *Implicit Learning: Theoretical and empirical issues.* Hove, UK: Lawrence Erlbaum.

Booth, W. (1996) Variation on a Theme Park: Key West in Age of Absolute Simulation. *International Herald Tribune,* April 20, p.7.

Bullinaria, J. & Chater, N. (1995). Connectionist Modelling: Implications for Cognitive Neuropsychology. *Language and Cognitive Processes*. 10(3/4), 227-264.

Burtt, E. (1932). *Metaphysical Foundations of Modern Physical Science*. London: Routledge (1972 edition).

Cleeremans, A. (in press). Principles for implicit learning. In D. Berry (Ed.), *What is implicit about implicit learning*. Oxford, England: Oxford University Press.

Cleeremans, A. (1993). *Mechanisms of implicit learning: Connectionist models of sequence processing*. Cambridge, MA: MIT Press.

Cleeremans, A. & French, R. M. (1996). From Chicken-Squawking to Cognition: The role of Computational Modeling in Psychology. *Psychological Belgica*. (in press).

Cleeremans, A. & McClelland, J. (1991). Learning the structure of event sequences. *Journal of Experimental Psychology: General*. 120, 235–253.

Cleeremans, A., Servan-Schreiber, D., & McClelland, J. (1989). Finite State Automata and Simple Recurrent Networks. *Neural Computation*, 1, 372-381.

Chalmers, D. (1990). Syntactic transformations on distributed representations. *Connection Science*, 2, 53-62.

Clark, A. (1990). *Microcognition: Philosophy, cognitive science, and parallel distributed processing*. Cambridge, MA.: MIT Press.

Crick, F. (1989). The recent excitement about neural networks. *Nature*, 337, 129–132.

Dennett, D.C. (1991). *Consciousness explained*. London: The Penguin Press.

Dunn, J.C., & Kirsner, K. (1988). Discovering functionally independent mental processes: The principle of reversed association. *Psychological Review*, 95, 91–101.

Elman, J.L. (1990). Finding structure in time. *Cognitive Science*, 14, 79–211.

Estes, W.K. (1957). Toward a statistical theory of learning. *Psychological Review*, 7, 94–107.

Farah, M.J. (1994). Neuropsychological inference with an interactive brain: A critique of the "locality" assumption. *Behavioral and Brain Sciences*, 7, 43–104.

Feldman, J.A., & Ballard, D.H. (1982). Connectionist models and their properties. *Cognitive Science*, 6, 205–254.

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

Forster, M. (1988): "Unification, Explanation, and the Composition of Causes in Newtonian Mechanics." *Studies in the History and Philosophy of Science* 19: 55 - 101

French, R. (1995). *The Subtlety of Sameness*. Cambridge, MA: The MIT Press.

French, R. (1990) Subcognition and the Limits of the Turing Test. *Mind*. 99(393), 53-65.

Gilovitch, T. (1991). The hot hand and other illustrations of everyday life. *The Wilson Quarterly*, 15(2), 52.

Goldstone, R.L., & Krushke, J.K. (1994). Are rules and instances subserved by separate systems? *Behavioral and Brain Sciences*, 17, 405.

Golomb, D., Wang, X. & Rinzel, J. (1994). Synchronization Properties of Spindle Oscillations in a Thalamic Reticular Nucleus Model. *Journal of Neurophysiology*. 72. 1109-1126.

Harnad, S. (1985). D. O. Hebb: Father of Cognitive Psychobiology. http://www.princeton.edu/~harnad.

Hebb, D. O. (1949). *The Organization of Behavior*. New York, NY: John Wiley and Son, Inc.

Hofstadter, D. and the Fluid Analogies Research Group (1995). *Fluid Concepts and Creative Analogies*. New York: Basic Books, Inc.

Hofstadter, D. (1979) *Gödel, Escher, Bach: an Eternal Golden Braid*. New York: Basic Books, Inc.

Laird, J.E., Rosenbloom, P.S., & Newell, A. (1985). Towards chunking as a general learning mechanism. (Tech. Rep. No. CMU-CS-85-100). Pittsburgh, PA: Carnegie Mellon University, School of Computer Science.

Marr, D. (1982). *Vision.* San Francisco: Freeman.

McClelland, J.L., Rumelhart, D.E., & Hinton, G. E. (1986). The appeal of Parallel Distributed Processing. In D.E. Rumelhart & J.L. McClelland (Eds.), *Parallel Distributed Processing: Explorations in the microstructure of cognition* (Vol. 1, pp. 3–44), Cambridge, MA: MIT Press.

McCloskey, M. (1991) Networks and theories: The place of connectionism in cognitive sciene. *Psychological Science*, 2, 387-395.

Miles, R., Traub, R., & Wong, K. (1988) Spread of Synchronous Firing in Longitudinal Slices from the CA3 region of the hippocampus. *Journal of Neurophysiology*. 60, 1481-1496.

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

Newell, A., Shaw, J.C., & Simon, H.A. (1957). Empirical explorations with the Logic Theory Machine: A case study in heuristics. In E.A. Feigenbaum & J. Feldman (Eds.), 1963. *Computers and Thought*. New York: McGraw-Hill, 109-233.

Newell, A. (1990). *Unified theories of cognition*. Cambridge, MA: Harvard University Press.

Nissen, M.J. & Bullemer, P. (1987). Attentional requirements of learning: Evidence from performance measures. *Cognitive Psychology*, 19, 1–32.

Patterson, K.E., & Marcel, A.J. (1977). Aphasia, dyslexia, and the phonological coding of written words. *Quarterly Journal of Experimental Psychology*, 29, 307–318.

Plaut, D.C. (1995). Double dissociation without modularity: Evidence from connectionist neuropsychology. *Journal of Clinical and Experimental Neuropschology*, 17, 291–326.

Plaut, D.C., & Shallice, T. (1993). Deep dyslexia: A case study of connectionist neuropsychology. *Cognitive Neuropsychology*, 10, 377–500.

Pollack, J. (1989). Implications of recursive auto associative memories. In D. Touretzky (Ed.), *Advances in Neural Information Processing Systems*. San Mateo, CA: Morgan Kaufman. 527-536.

Rochester, N., Holland, J. H., Haibt, L.H. & 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 Theory IT-2*, 80-93. Reprinted in J.A. Andreson & E. Rosenfeld (Eds.), 1990. *Neurocomputing: Fondations of research*. Cambridge MA: MIT Press, 68-79.

Rosenbloom, P., Newell, A., & Laird, J. (1990) Toward the knowledge level in Soar: The role of the architecture in the use of knowledge. In K. Van Lehn (Ed.), *Architectures for Intelligence*. Hillsdale, N.J.: Erlbaum.

Rumelhart, D.E., Hinton, G.E, & Williams, R.J. (1986). Learning representations by back-propagating errors. *Nature*, 323, 533–536.

Rumelhart, D.E., and McClelland, J.L. (1986). On learning the past tense of english verbs. In J.L. McClelland & D.E. Rumelhart (Eds.), *Parallel Distributed Processing: Explorations in the microstructure of cognition* (Vol. 2, pp. 216–271) Cambridge, MA: MIT Press.

Samuel, A.L. (1959). Some studies in machine learning using the game of checkers. In E.A. Feigenbaum & J. Feldman (Eds.), 1963. *Computers and Thought*. New York: McGraw-Hill, pp. 71-105.

Seidenberg, M., & McClelland, J.L. (1989). A distributed, developmental model of word recognition and naming. *Psychological Review*, 96, 523-568.

Seidenberg, M. (1993). Connectionist model s and cognitive theory. *Psychological Science*, 4, 228–235.

Sejnowski, T. and Rosenberg, C. (1987). Parallel networks that learn to pronounce english text, *Complex Systems*, 1, 145–168.

Servan-Schreiber, E., & Anderson, J.R. (1990). Learning artificial grammars with competitive chunking. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 16, 592-608.

Smolensky, P. (1988). On the proper treatment of connectionism. *Behavioral and Brain Sciences*, 11, 1–74.

Thomas, E. & Wyatt, R. (1995). A computational model of thalamocortical spindle oscillations. *Mathematics and Computers in Simulation*, 40, 35-69.

van Gelder, T. (1995) Modeling, connectionist and otherwise. *Proceedings of the Swedish Conference on Connectionism*. Lawrence Erlbaum Associates (in press).

Weaver, M. (1993) An active-symbol connectionist model of concept representation and concept learning. Unpublished doctoral dissertation, Computer Science Department, University of Michigan.