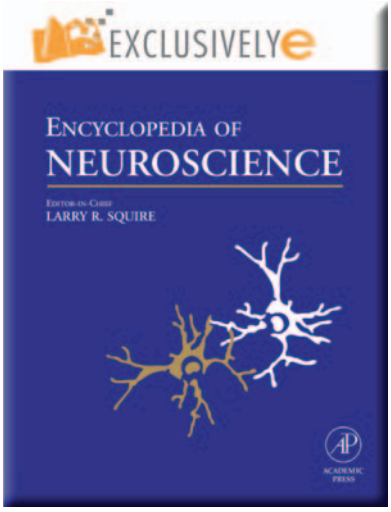


Rogers T T (2009) Connectionist Models. In: Squire LR (ed.) Encyclopedia of Neuroscience, volume 3, pp. 75-82. Oxford: Academic Press.

Connectionist Models

T T Rogers, University of Wisconsin–Madison, Madison, WI, USA

Connectionism is a theoretical framework for cognition whose principal tenets are (1) that all cognitive phenomena arise from the propagation of activation among simple neuronlike processing units and (2) that such propagation is mediated by weighted synapselike connections between units. Connectionist theories are typically instantiated as computer models, that is, computer programs that simulate how activation propagates through the system of interconnected units specified by the theory. Such models have profoundly influenced virtually every subdomain of cognitive science: perception and attention; word recognition, reading, derivational morphology, and other aspects of language; episodic, semantic, and short-term memory; action and other forms of sequential processing; executive function and cognitive control; many aspects of cognitive development; and even emotion. Many researchers, however, consider the connectionist framework to be incommensurable with the mainstream view that the human mind is a symbol-processing system and, for this reason, consider connectionist theories to be fundamentally flawed, or at least insufficient to serve as a general framework for cognition. This article considers the historical roots of connectionism, reviews key aspects of the approach, and addresses some of the criticisms it faces.

History

The roots of connectionism lie in the mid-1940s, when Warren McCullough and Walter Pitts showed how individual neurons could be viewed as simple computing devices, assuming that:

1. Neurons can assume two states (on or off, that is, 1 or 0).
2. Each neuron has a fixed threshold above which it assumes the on state.
3. A given neuron's activation state is influenced by the other neurons from which it receives connections.
4. The influence of a sending neuron on the receiving neuron depends on the value of the synapse or weight connecting them.

McCullough and Pitts demonstrated that individual neurons could compute the basic two-bit logic functions (AND and OR) and that networks of such neurons could compute more complex logic functions (e.g., exclusive-OR (XOR)).

At around the same time, Donald Hebb had become interested in the cellular basis of learning and memory, and he suggested a mechanism for changing the efficacy of a synapse to support a simple form of associative learning. Specifically, he suggested that, "When an axon of cell A is near enough to excite B and repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A's efficiency, as one of the cells firing B, is increased" (Hebb, 1949). This insight – that correlated activity in a system of neurons can produce synaptic change – provided a basis for understanding the neural mechanisms that support learning and memory.

These contributions provided the foundations for the connectionist enterprise. Like other approaches, the connectionist framework assumes that cognitive systems are information processing systems that take in information via sensory organs, transform the information to form internal representations of the environment, and from these representations generate outputs in the form of overt behaviors. In contrast to some other approaches, the connectionist framework posits that the functional elements that support cognitive information processing are something like the abstract neurons described by McCullough and Pitts – units that take on activation states depending on their inputs and transmit these states to other units via weighted connections. It also posits that the principal mechanism for learning and memory is something like that proposed by Hebb – learning is effected through changes to the weights that connect processing units; such weight changes are partly (although not necessarily completely) driven by patterns of correlated activity across units in the system; and such correlated patterns in turn depend on experience.

These foundational ideas were developed in the 1950s and 1960s, largely through the work of Frank Rosenblatt, who introduced the perceptron, a form of neural network similar in some respects to those described by McCullough and Pitts. Rosenblatt derived a learning algorithm (a rule for changing the weights between units) that allowed the perceptron to solve a much wider range of learning problems than the Hebb learning rule. Rather than changing weights in proportion to the correlation between connected units, the perceptron learning rule minimizes the error generated by an input. That is, for each input, the algorithm compares what the network actually produced to the pattern it should have produced (the target pattern) and adjusts the values of each

weight so as to reduce the difference between these (the error, sometimes referred to as delta because it describes a difference between two terms). This delta-rule learning algorithm thus requires a teaching signal (the network must be told what pattern it is to generate in response to each input), but given this signal it can learn input–output mappings that are well beyond the Hebb rule.

Rosenblatt died tragically in a boating accident in 1969 at the age of 41. The same year, in a book called *Perceptrons*, Marvin Minsky and Seymour Papert published a forceful critique of perceptron-based theories of cognitive function. The book showed that networks of simple perceptrons were incapable of learning to compute nonlinearly separable mappings between input and output; for instance, they could not learn second-order logical functions such as XOR. Largely on this basis, the authors argued that perceptrons could not, in principle, provide a basis for understanding human cognition. This book, coupled with the intensive lobbying (by Minsky) of various funding agencies and other scientists, seriously dampened general interest in and funding for neural network modeling for the next 10 years, although important advances were made during this period by a number of scientists working independently.

The Modern Period

The 1980s witnessed a renaissance of interest in connectionism, for three principal reasons. The first was the publication of some landmark papers that set aside the difficult questions about learning raised by Minsky and Papert but that used connectionist principles and implemented computer models to understand a range of cognitive phenomena. Perhaps most influential was the interactive activation and competition model of word recognition, which accounted for a very wide range of behavioral phenomena in the domain and resolved several important puzzles pertaining to lexical processing (see **Figure 1**). Among other things, this model introduced the notions that the component processes involved in word recognition proceed in parallel and not in serial; that processing is cascaded, so that a later process can begin to operate on the outputs of earlier processes before the earlier processes have themselves completed; and that bottom-up and top-down processes continually interact throughout processing. These notions were antithetical to the then-dominant view that cognitive processes were carried out by informationally encapsulated modules whose operation proceeded largely in serial.

The second major development was the discovery of a solution to the perceptron-learning problems raised by Minsky and Papert. The discovery was made independently by various investigators, but its importance to cognitive science was made clear in an influential article by Rumelhart, Hinton, and Williams. The key insight concerned the nature of the transfer function that governed how a single unit sets its activation state in response to inputs. Previous investigators had, for simplicity's sake, assumed either a linear transfer function or a step function; and the limited the scope of the perceptron-learning rule was a direct consequence of these choices. Rumelhart and colleagues investigated a logistic transfer function which (1) is nonlinear like a step function but (2) varies smoothly with net input so that it is differentiable (and, in fact, has a comparatively simple derivative; see **Figure 2(b)**). Because the transfer function is nonlinear, groups of units arranged in series are capable of computing the nonlinearly separable mappings that cannot be learned in a multilayered perceptron with linear units. Because

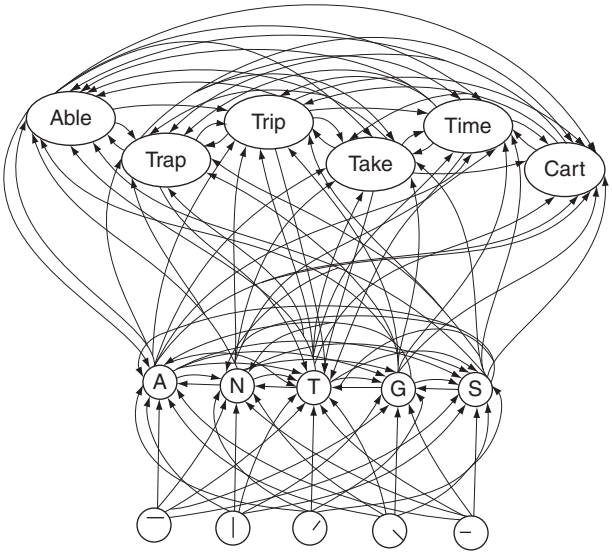
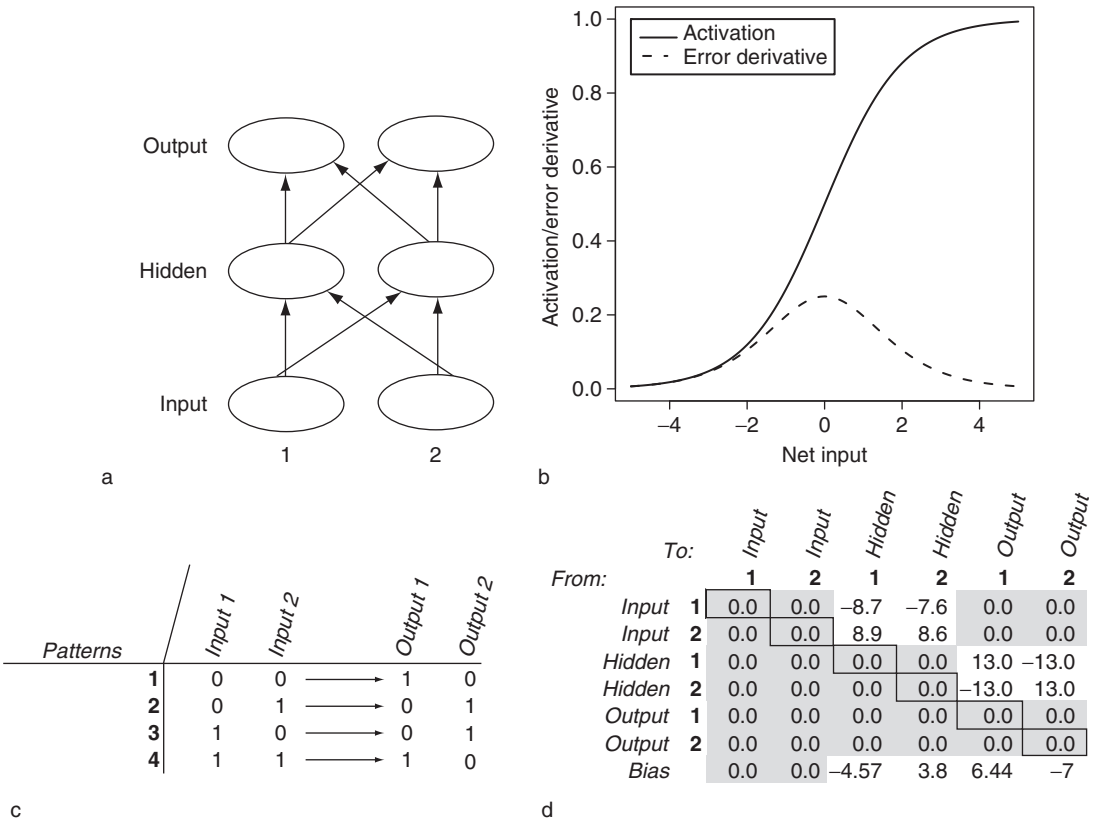


Figure 1 The interactive activation model of word recognition. The model consists of feature detectors organized hierarchically so that those toward the bottom represent simple visual features, those in the middle represent individual letters, and those at the top represent individual words. The visual appearance of an individual word activates basic features at the bottom, which pass their activation up through the hierarchy via weighted connections. Features excite the letters with which they are compatible and inhibit the letters with which they are incompatible. Individual letters inhibit one another, excite word units with which they are compatible, and inhibit word units with which they are incompatible. The activation of word-level units can feed back down to influence letter-level units, inhibiting incompatible letters and further exciting compatible units. Processing of a given input thus depends on the interactive activation of representations at all levels of the hierarchy. From McClelland JL and Rumelhart DE (1981) An interactive activation model of context effects in letter perception, Part 1: An account of basic findings. *Psychological Review* 88: 375–407.



c

Patterns	Input 1	Input 2		Output 1	Output 2
1	0	0	→	1	0
2	0	1	→	0	1
3	1	0	→	0	1
4	1	1	→	1	0

d

From: \ To:		Input **1**	Input **2**	Hidden **1**	Hidden **2**	Output **1**	Output **2**
Input	**1**	0.0	0.0	–8.7	–7.6	0.0	0.0
Input	**2**	0.0	0.0	8.9	8.6	0.0	0.0
Hidden	**1**	0.0	0.0	0.0	0.0	13.0	–13.0
Hidden	**2**	0.0	0.0	0.0	0.0	–13.0	13.0
Output	**1**	0.0	0.0	0.0	0.0	0.0	0.0
Output	**2**	0.0	0.0	0.0	0.0	0.0	0.0
Bias		0.0	0.0	–4.57	3.8	6.44	–7

Figure 2 Exclusive- or (XOR) mapping: (a) architecture of a six-unit network for computing the exclusive- or (XOR) mapping; (b) logistic transfer function indicating how a unit's activation state changes with its net input; (c) a model environment containing patterns for the XOR learning problem; (d) one-weight matrix learned by the backpropagation learning rule. In (a), each oval indicates a neuronlike processing unit, and each arrow indicates a synapselike connection between units. In (b), a given receiving unit computes its net input by taking the dot product of the activations of sending units from which it receive projections times the values of the interconnecting weights. The dotted line indicates the derivative of the activation function, which can be used by the backpropagation learning rule to adjust the magnitudes of weights in the network so as to reduce error at the output layer. In (c), to process each pattern, the input units in (a) are set to the values indicated in the table (1 or 0), and this activation spreads forward in the network as determined by the values of the interconnecting weights and the transfer function. The activations of the two output units are then compared to the desired outputs shown in the table; the difference is converted to a measure of error, and the derivative of this error with respect to each weight is computed in a backward pass from outputs toward inputs. Each weight is adjusted by a small amount so as to reduce error at the output. In (d), the gray cells indicate logically possible connections between neurons that are not present in the network architecture, so their values are fixed to zero. The line labeled bias indicates each unit's intrinsic bias, which is a constant added to the unit's net input. When each of the input patterns in (c) is presented to a network with these weight values, outputs closely approximate the desired outputs also listed in (c).

the function is differentiable, it is possible to compute how each weight in such a network should be adjusted to improve performance. It is therefore possible to train a network of such units by (1) presenting an input and computing the forward propagation of activation from inputs toward outputs; (2) computing a measure of error, that is, a measure of the discrepancy between the observed and desired (target) output; and then (3) computing the derivative of the error with respect to each weight in the network in a backward pass from outputs toward inputs, and using these to adjust each weight in the network so as to reduce the error at the output. Networks trained with this backpropagation learning rule can, in principle, learn any input-output mapping, so long as there is an intervening layer of at least two units between inputs and outputs.

The third watershed event was the publication in 1986 of *Parallel Distributed Processing*, a two-volume work that laid out a comprehensive rationale for basing cognitive theories on principles derived from neural computation, reviewed the methodological state of the art, reported models of cognitive function across a remarkably broad variety of cognitive domains, and linked these models to core principles derived from theories of neural information processing. The two volumes highlighted some of the key differences between the then-dominant mind-as-computer approach to cognition and a neurally inspired approach. For instance, whereas computers

employ blindingly fast serial processors to achieve their computational power, individual neurons operate on an orders-of-magnitude slower time scale but are massively parallel. Despite the relatively slow computations achieved by a single neuron, neural systems achieve remarkably rapid and accurate performance in tasks that seriously challenge even the fastest serial computers (e.g., speech recognition). This suggests some fundamental difference in the way that parallel and serial systems process information; and if the mind is a product of a massively parallel processing system, it follows that the endeavor to understand cognition as the product of a serial symbol-processing system may not be optimal.

Basic Properties of Connectionist Networks

Connectionist models have seven general properties: (1) a set of units, (2) an activation state, (3) a weight matrix, (4) an input function, (5) a transfer function, (6) a learning rule, and (7) a model environment.

A Set of Units

Units are abstract neurons, simple processors that set their activation level by combining incoming signals and then transmit their activation state to other units through weighted connections. Some models construe individual units as representing particular theoretical constructs – for instance, explicit stimulus properties (such as colors, shapes, individual letters, or phonemes) or even individual words, objects, or concepts. Other models suggest that such theoretical constructs are best represented as patterns of activity across collections of units, so the activation of any individual unit has no directly interpretable significance. Figure 2 (a) shows the architecture of a feed-forward model for computing the XOR function.

An Activation State

At any given point in time, every unit in a network has a certain activation state. The state of the entire network, then, may be conceived as a one-dimensional vector, with each element corresponding to a single unit and containing that unit's current activation. When processing any given input pattern, the state of the XOR network is represented with a vector of six elements.

A Weight Matrix

Units can influence one another's activation via synapselike connections, or weights. Weights have direction; they project from a sending unit i to a receiving unit j. The weights in a network with n units can thus be conceived as an $n \times n$ matrix, with each element indicating the value of the weight projecting from one neuron to another, elements below the diagonal coding weights projecting in one direction (i.e., from i to j) and elements above the diagonal coding weights projecting in the reverse direction (i.e., from j to i). Most networks also include a vector of bias weights, that is, a vector then includes one constant for each unit in the network that reflects that unit's base activation in the absence of input. The architecture of the network (the arrangement of units into groups or layers) constrains which units project to which other units; where no connection exists between two units, the value in the weight matrix is fixed to zero. Figure 2(d) shows the matrix for a configuration of weights that solve the XOR mapping; gray cells in the matrix indicate values fixed to 0 because no connection exists between the corresponding units (A in Figure 1).

An Input Function

At any given time, a receiving unit j receives input from some set of sending units via the intermediating weight. The input function associated with a network determines how these inputs are combined to provide a net input to the receiving unit. The most frequent input function is simply the dot product of the vector of sending activations and the weight vector; that is, for each sending unit, one computes the product of the unit's activation and the value of the weight projecting to the receiving unit. These products are then summed across all sending units, and the result is the receiving unit's net input.

A Transfer Function

The transfer function determines how a unit's current net input influences its activation state. As previously mentioned, the logistic transfer function (a sigmoid bounded at 0 and 1) is very commonly used due to its desirable computational properties; but linear functions, step functions, radial basis functions, and other choices all appear in the literature. Models can also vary in how the transfer function integrates time. In many models, units update their activation instantaneously, but it is possible to gradually update the unit's state so as to simulate temporally extended network behaviors. Figure 2(b) shows the logistic transfer function and its derivative.

A Learning Rule

Although the weights in a connectionist model can sometimes be set by hand (i.e., specified by the

modeler), a key aspect of their appeal is their ability to discover useful weights as the consequence of applying some learning rule, that is, some rule for determining how weights should change as the consequence of processing an input pattern. There now exist a variety of learning rules, including many variants of the Hebb rule, in which weights are adjusted on the basis of correlated activity between pairs of units; various forms of error-correcting learning, such as the delta rule and backpropagation; and unsupervised learning rules that come to categorize, or cluster, their inputs without any explicit indication of what the desired output should be. There remains considerable controversy regarding the significance or utility of particular learning rules, the principal issue being an apparent trade-off between computational power and biological plausibility. Specifically, backpropagation remains the most powerful learning rule yet discovered, but it makes use of error signals that, at least at first blush, appear to have no analog in real brains. In contrast, variants of the Hebb rule closely reflect changes to synaptic potentiation in real brains; consequently, such rules are sometimes viewed as more biologically plausible, but they are also known to be fairly limited in the kinds of computations they can learn to perform.

A Model Environment

Because, in the connectionist view, all information processing involves the propagation of activation among units, it follows that, in any given task, both the stimulus and the behavior must be represented as patterns of activation across subsets of input and output units. The theorist must explicitly specify how such inputs and outputs are coded; in models of reading, for instance, the modeler must specify how the visual appearance of a word form and the motor act of articulating a word are to be coded as patterns of activity across units. The model environment consists of a collection of input–output pairs that constitute the mappings to be learned. Because a model's behavior is strongly influenced by the structure of its inputs and outputs, decisions about how these are best represented are often key to the theory that a given model is intended to exemplify. The training environment for the XOR problem is shown in Figure 2(c).

Appeal of Connectionist Models

Models with the seven properties just described provide an appealing basis for understanding aspects of cognition that seem especially challenging from traditional mind-as-computer approaches to cognition. For instance, such models exhibit:

1. Content-addressable memory. Connectionist models are good at pattern completion. Given some partial information about a previously learned pattern, the propagation of activation through a connectionist net can reinstantiate the full pattern; and moreover, any part of the full pattern can later be used as the cue for retrieval. This is also clearly true of human memory, but it is not naturally achieved in computer systems, in which a given data record can be retrieved only if one knows the correct address (i.e., the physical location on the storage medium where the record begins).
2. Generalization. In most connectionist networks, all patterns are processed through the same network of units and weights. Consequently, new inputs that share structure with previously learned items tend to generate similar outputs to the previously learned items. This ability to generalize past learning to new situations is critical to connectionist accounts of memory and language.
3. Graceful degradation. As any software engineer knows, computer programs are extremely brittle – tiny flaws in a program can completely destroy its ability to function. In contrast, connectionist networks exhibit graceful degradation. When weights or units are removed from the system, such networks can often generate outputs that are completely or at least partially correct. The human cognitive system also appears to exhibit graceful degradation, in that brain damage rarely if ever produces an all-or-nothing pattern of cognitive impairment. Furthermore, because the elements of a connectionist model can correspond, at least in principle, to particular brain regions, it is possible to investigate how brain damage in a particular locus should influence behavior under a given theory. Thus, connectionist models offer a valuable tool for the development of neuropsychological theories.
4. A mechanism for cognitive change. More traditional information-processing approaches are often used to characterize a child's mental processes at different stages of development, but under such approaches it is often unclear what mechanism subserves the transition from one stage to another. Connectionist models provide explicit hypotheses about mechanisms of change, in the form of the learning rule used to train the model, explicit assumptions about the nature of the developing child's experience, and explicit assumptions about how maturational change affects the architecture of the neural network

supporting the behavior. So, such models also offer a means of generating explicit hypotheses about mechanisms underlying cognitive development.

Symbolic versus Connectionist Approaches

Connectionist theories are sometimes viewed as providing a fundamentally different approach to cognition than the more traditional symbol-processing approach. Symbol-processing theories assume that the mind is a computational system consisting of two basic elements: (1) symbols, that is, structured representations that stand for things in the world, and (2) rules or operations, that is, algorithms for manipulating, combining, and responding to symbols. For instance, to generate the past-tense form of a verb such as 'look' might require, under a symbol-processing view, two symbols – one standing for the word 'look' and one standing for the English past-tense suffix '-ed' – and a rule that says, "'IF the goal is to generate the past tense from the present tense, THEN attach the past-tense suffix to the present-tense form." Applying the rule to the symbols then generates the correct past-tense form, 'looked.' Moreover, such a symbolic approach has a certain generative power; after learning a new present-tense verb (e.g., 'mave'), it is possible to infer the past-tense form ('maved') simply by applying the past-tense rule to the newly learned symbol.

Because symbol-processing systems are generative (i.e., they can produce an infinite variety of new representations and behaviors from a finite set of starting elements), many scientists believe that they must underlie much of human cognition, which is also generative. Connectionist models, in contrast, are thought by some to be capable of showing, at best, only a limited range of generativity. The current debate concerning the merits of connectionist approaches often hinge on these issues, which are best illustrated by an example.

Consider that in English, approximately 80% of verbs have a past tense formed by simply adding '-ed' to the present-tense form. The remaining 20% have past-tense forms that may be somewhat related to the present-tense form (e.g., 'give' → 'gave') or may even be completely unrelated to it (e.g., 'go' → 'went'). How do people come to learn which past-tense forms to produce from a given present-tense verb?

One commonsense hypothesis is that the present- and past-tense forms of familiar verbs are linked together in memory, by virtue of prior learning. Several empirical facts about language appear, however, to challenge this view. For instance, (1) people can reliably generate past-tense forms for novel verbs, and because these have never before been encountered, the behavior cannot depend on simply remembering the correct form; (2) children often generate incorrect forms that they are unlikely ever to have heard (e.g., producing 'goed' as the past tense of 'go'); and (3) a similar kind of behavior is observed following some kinds of brain damage, and it is absurd to imagine that brain damage causes someone to suddenly remember some past-tense form they have never before experienced or produced.

Such phenomena suggest that past-tense formation is subserved by something other than an association in memory between present- and past-tense forms. Symbol-processing theories propose that this something other is a rule like the one articulated earlier. In this view, the rule is used to generate past-tense forms for the majority of verbs, including novel verbs. Exceptions to the rule are stored in a separate lookup system or lexicon; and there may be additional rules for handling such exceptions (e.g., IF the present tense is listed in the lexicon, THEN look up the past-tense form in the lexicon; OTHERWISE use the rule to generate the past-tense form). Such a proposal nicely explains the empirical facts: because novel verbs such as 'mave' are not stored in the lexicon, the rule is used to generate a lawful past-tense form; children, on initially acquiring the rule, may mistakenly overapply it to exception words such as 'go' (thus accounting for their production of forms such as 'goed' that they have never encountered); patients who, after brain damage, generate forms such as 'goed' or 'drived' may have an intact rule system but a degraded exception system and so apply the rule to previously known exceptional forms.

It turns out, however, that the associative-learning account is not as limited, nor is the symbolic approach as powerful, as we might initially suspect. Connectionist networks trained to generate past-tense from present-tense forms (with real English words) actually exhibit all of the behaviors listed. The reason has to do with the remarkable ability of such models to generalize from previous experience. For instance, although such a network is never trained with a nonword such as 'mave', it does, during training, encounter real verbs with a similar structure: 'save', 'pave', 'rave', and so on. Most of these words form their past tense by adding '-ed' to their present-tense forms; and because this is so, new items such as 'mave' tend to produce a similar output. Such systematic tendencies in the language can also produce patterns of overregularization early in network training, as when children produce 'goed' instead of 'went', or when the knowledge in the network is perturbed by removing connection weights or units (as a model analog of brain damage).

Moreover, connectionist models so trained explain other observations that challenge the symbolic approach. For example, although exception words do not, like most verbs, simply take -'ed' in their past-tense form, there are some very systematic tendencies among the exceptions – consider 'weep' → 'wept', 'keep' → 'kept', 'sleep' → 'slept', and so on. These formations are not just consistent with one another; they are also quite similar to the past-tense form that would be generated if the rule were applied – the only difference is that, for these words, the long vowel is shortened in the past tense. Such observations are difficult to explain under the rules-and-symbols view. If exceptions are generated by a lookup table, why are past-tense forms so systematic across different subgroups of exceptions? And, if the rule and exception systems are functionally independent of one another, why are so many exceptional forms so similar – often differing in just one or two phonemes – to the regular forms that would be generated under the rule system?

Connectionist theories of past-tense formation explain such observations by suggesting that past-tense forms for both regular and exception words are generated through the same set of units and weights. The weights are shaped by systematic structure in the language, via the operation of a simple learning rule such as backpropagation. Such learning mechanisms excel at discovering weights that simultaneously capitalize on systematic input–output relationships, where they exist, and tolerate mild or even strong deviations from such regularities where necessary. Thus, although mappings such as 'keep' → 'kept' are in some sense exceptions, they still share considerable structure with the regular verbs and with one another; so these exceptional forms actually benefit substantially by being processed through the same system that handles the regular verbs. And, the connectionist account predicts empirical findings that are difficult to reconcile with the rules-and-symbols view. For example, when presented with a new nonsense verb such as 'neep', people are actually more likely to generate an irregular past-tense form (e.g., 'nept') than the regular form (i.e., 'neeped'), even though, according to the rule-based view, they should always apply the past-tense rule to all novel forms.

Past-tense formation thus exemplifies a domain in which (1) there exist very intuitive and empirically fleshed-out reasons for believing that behavior must be supported by rules and symbols but for which (2) simulations with connectionist models demonstrate that there exist alternative explanations for the same behaviors, within which there are no analogs to rules and symbols. Similar issues and controversies arise in the study of reading, language comprehension, word and object recognition, and categorization and semantics. In all these domains, connectionist theories usually explain behavior without reference to rules and symbols, and consequently such theories are sometimes viewed as challenging the view that human minds are symbol-processing systems. Other domains of human behavior – including, arguably, language production, reasoning, problem solving, and mathematics – seem to many to require a symbolic approach; consequently, challenges to symbolic explanations of even relatively simple phenomena (such as past-tense formation or single-word reading) may be viewed either as flawed in principle (because it is difficult to see how such accounts can be extended to higher-level cognition) or as undermining the foundational assumptions of cognitive science.

There are, however, other less dialectical ways of conceiving the relationship between symbolic and connectionist approaches. One view is that human cognition is subserved by some hybrid of connectionist and symbol-processing systems. For instance, recent symbolic accounts of past-tense formation suggest that the exception system is supported by a connectionist-like associative memory (thus explaining the systematicity among exceptions) but that there also exists a separate and independent rule system for handling regular items that bear little resemblance to previously learned forms.

Another view is that symbolic and connectionist approaches account for human behavior at different levels of description. Symbolic approaches may, for instance, provide valid descriptions of human behavior much as a flowchart captures the operation of a computer program. The flowchart is a useful description of the program, even though it yields no insight into the causal mechanisms by which the underlying hardware carries out the resulting computations. Connectionist theories, in contrast, may capture a level of description that explains how cognitive behaviors that, roughly speaking, look symbolic, can arise from mechanisms similar to those that govern neural information processing.

Biological Plausibility

Connectionist models are sometimes criticized for making implausible claims to biological faithfulness. For instance, it is not clear to what the units in many connectionist models are intended to correspond: single neurons, cortical columns, or whole populations of neurons. Similarly, single units can be used to represent whole words, objects, or other entities that, in the brain, are probably represented by widely distributed patterns of activity across thousands of neurons; unit activations are represented with single

real-valued numbers, whereas actual neurons transmit information via discrete all-or-nothing action potentials; weights in connectionist networks often can take any real positive or negative value, whereas in the brain, some synapses are exclusively inhibitory and others are exclusively excitatory; in connectionist models, different units are often treated as functionally and structurally equivalent to one another, whereas in the brain there exist many different kinds of neurons, which may in turn participate in various different cytoarchitectonic structures; and so on. For such critics, claims that connectionist models capture important aspects of neural information processing may seem far-fetched.

One response to this criticism is to point out that the utility of any model system lies in abstracting away from the many complex details of the natural system, so as to investigate those factors that, under a given theory, are necessary to explain the relevant phenomena. Thus, it is certainly possible to construct neurally faithful computer simulations in which there exist multiple different kinds of neurons, connected via different kinds of synapses, organized into cortical columns, and projecting to one another via neuroanatomically faithful tracts. Should the theorist demonstrate that such a simulation replicates the behavior of interest, however, it would be comparatively difficult to understand why this should be so because the model is nearly as complex as the system itself. Connectionist models may be viewed as making simplifying assumptions about the details of the underlying neural systems so as to allow the theorist to focus on a relatively circumscribed set of factors of explicit theoretical interest.

It is also worth noting that the investigation of the properties of abstract computational systems can lead to important neuroscientific discoveries, even when the computational systems are not intended as faithful models of neural systems. For instance, in 1988 Sutton described an algorithm for reward-based learning of sequential structure (temporal-difference learning) in work that was purely computational (i.e., not guided by neuroscience). Nevertheless, it was subsequently discovered that dopamine signals in cortex, striatum, and midbrain closely track the reward signal generated by the algorithm, and it now appears as though some forms of learning may be supported by neural mechanisms that effectively implement temporal-difference learning. So, insights gained through computer simulations, even when they may seem divorced from what we currently know about neural processing mechanisms, can promote new insights into how neural systems function.

See also: Cognition: An Overview of Neuroimaging Techniques; Cognitive Neuroscience: An Overview; Connectionist Models of Language Processing; Executive Function and Higher-Order Cognition: Computational Models; History of Neuroscience: Early Neuroscience; Recognition Memory; Word Recognition; Word Learning.

Further Reading

Churchland PS and Sejnowski TJ (1990) Neural representation and neural computation. *Special Issue: Action Theory and Philosophy of Mind. Philosophical Perspectives* 4: 343–382.

Cohen JD, Braver TS, and O'Reilly R (1996) A computational approach to prefrontal cortex, cognitive control and schizophrenia: Recent developments and current challenges. *Philosophical Transactions: Biological Sciences* 351(1346): 1515–1527.

Elman JL (1990) Finding structure in time. *Cognitive Science* 14: 179–211.

Elman JL, Bates EA, Karmiloff-Smith A, Johnson M, Parisi D, and Plunkett K (1996) *Rethinking Innateness: A Connectionist Perspective on Development.* Cambridge, MA: MIT Press.

Fodor JA and Pylyshyn ZW (1988) Connectionism and cognitive architecture: A critical analysis. *Cognition* 28: 3–71.

Hebb DO (1949) *The Organization of Behavior.* New York: Wiley.

Joanisse MF and Seidenberg MS (1999) Impairments in verb morphology after brain injury: A connectionist model. *Proceedings of the National Academy of Sciences of the United States of America* 96: 7592–7597.

Marcus GF (2001) *The Algebraic Mind.* Cambridge, MA: MIT Press.

McClelland JL and Patterson K (2002) "Words or rules" cannot exploit the regularity in exceptions. *Trends in Cognitive Sciences* 6: 464–465.

McClelland JL and Rumelhart DE (1981) An interactive activation model of context effects in letter perception, Part 1: An account of basic findings. *Psychological Review* 88: 375–407.

Pinker S and Ullman MT (2002) The past and future of the past tense. *Trends in Cognitive Sciences* 6: 456–463.

Plaut DC, McClelland JL, Seidenberg MS, and Patterson K (1996) Understanding normal and impaired word reading: Computational principles in quasi-regular domains. *Psychological Review* 103: 56–115.

Rogers TT and McClelland JL (2004) *Semantic Cognition: A Parallel Distributed Processing Approach.* Cambridge, MA: MIT Press.

Rumelhart DE, McClelland JL, and the PDP Research Group (1986) *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations.* Cambridge, MA: MIT Press.

Rumelhart DE, McClelland JL, and the PDP Research Group (1986) *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 2: Psychological and Biological Models.* Cambridge, MA: MIT Press.