WHAT KNOWLEDGE MUST BE IN THE HEAD IN ORDER TO ACQUIRE LANGUAGE?

William Bechtel
Department of Philosophy
Georgia State University

1. Localizationist Dangers in the Study of Language

Many studies of language, whether in philosophy, linguistics, or psychology, have focused on highly developed human languages. In their highly developed forms, such as are employed in scientific discourse, languages have a unique set of properties that have been the focus of much attention. For example, descriptive sentences in a language have the property of being "true" or "false," and words of a language have senses and referents. Sentences in a language are structured in accord with complex syntactic rules. Theorists focusing on language are naturally led to ask questions such as what constitutes the meanings of words and sentences and how are the principles of syntax encoded in the heads of language users. While there is an important function for inquiries into the highly developed forms of these cultural products (Abrahamsen, 1987), such a focus can be quite misleading when we want to explain how these products have arisen or the human capacity to use language. The problem is that focusing on its most developed forms makes linguistic ability seem to be a *sui generis* phenomenon, not related to, and hence not explicable in terms of other cognitive capacities. Chomsky's (1980) postulation of a specific language module equipped with specialized resources needed to process language and possessed only by humans is not a surprising result.

The strategy of identifying a specific component within a system and assigning responsibility for one aspect of the system's behavior to that component is a common one in science. Richardson and I (Bechtel and Richardson, 1992) refer to this as *direct localization*. To see that direct localization is not a strategy unique to language studies or to explaining cognitive functions, we need only to consider the earliest attempts to explain fermentation. In the wake of Pasteur, many researchers doubted whether any chemical explanation of fermentation was possible. They thought that it was a unique capacity of yeast cells. However, in 1897 Eduard Buchner demonstrated that fermentation continued in extracts in which the whole cells had been destroyed. He then posited that there was a single enzyme, *zymase* that was responsible for the chemical process. Buchner's explanation soon proved to be inadequate as chemists recognized that fermentation was a many step process.

Since I am stressing the limitations of direct localization, I should also stress that it is often a fruitful first step in developing a more adequate understanding of how a complex system operates. Moreover, in fact, direct localizations are correct: there is a component in the system that performs the task that is assigned to it. The point to recognize then is that one still has not *explained* the ability until a decomposition is effected, for we do not understand how something is able to perform that activity. If the direct localization is correct, at least to a first approximation, then research typically proceeds at a lower level, where researchers try to take that component apart.

As research on fermentation continued, researchers developed a *complex localization* in which many different enzymes as well as coenzymes were identified as responsible for different components of the overall chemical transformation. The result, by the 1930s, was a complex model of interacting components that achieved the overall reaction of fermentation. Richardson and I have identified two heuristics that figured in this and other cases of developing complex localizations: the *decomposition* of a complex activity into simpler activities and the *localization* of responsibility for these activities in different

components. It would seem that the goals in trying to explain human linguistic abilities are similar: we want to know the various sorts of processes involved in language processing (task decomposition) and to identify the cognitive/neural components responsible for each (function localization).

In fact, such a program is in place in the study of language. A person's understanding of language is frequently decomposed into different kinds of knowledge: knowledge of syntax, semantics, pragmatics, etc. Psycholinguists attempt to identify component processes in human comprehension and production of language. A similar enterprise is pursued in artificial intelligence, where researchers are trying to develop parsers that can enable programs to extract useful representations of information from natural language inputs. Much of this work is very sophisticated and very impressive. But in this paper I want to raise a worry about the conceptualization of these projects and advance a different perspective from which to think about human linguistic ability. The worry can be focused by noticing that there is a step that must be performed even before one attempts a direct or a complex localization: one must identify a system that is responsible for the phenomenon. Richardson and I refer to this as identifying the *locus of control* for the phenomenon. In the case of language, it seems to many that this system is the mind/brain. The case for this seems to be overwhelming: humans comprehend and produce language, and the activities involved in doing this surely must be occurring inside their heads. But to recognize that this could be controversial, we only have to consider the approach against which Chomsky (1959) was reacting: Skinner's (1957) proposal to explain language using the tools of operant conditioning. Skinner's program was to minimize the contribution of the mind and to explain linguistic behavior in terms of environmental processes conditioning particular forms of behavior. The alternative to Chomsky that I will urge is, however, not Skinner's. My goal is not to discount the mind as playing a significant role in explaining linguistic capacities, but to suggest that linguistic ability be understood in terms of interactions between the mind and features of the environment.

Before beginning to develop my alternative proposal, let me note one of the consequences of localization of linguistic capacity in the mind. This is that the mind itself is construed as working on linguistic principles. Chomsky's transformational grammar employed procedures for manipulating strings of symbols that are composed in particular ways (often a tree structure is used to provide a more perspicuous representation). Psychologists such as George Miller were attracted to the idea that the mind might process language by performing such transformations, and more generally by the idea that the mind might operate by performing formal operations on strings of symbols. The availability of the computer, a device which can be interpreted as operating by performing formal operations on symbol strings, combined with Chomskyan linguistics in inspiring the development of the information processing tradition in psychology. The key to the information processing tradition is that the mind/brain is a representational device, and that it operates by performing operations upon the symbols that serve as its representations. These symbolic representations have much the character of linguistic representations, and Fodor (1975) in fact referred to the internal representational system of the mind as a *language of thought*.

For Fodor, the importance of the language of thought hypothesis is not just that the mind uses representations, but that these representations are structured in much the way that natural language representations are structured by principles of grammar. In fact, for him this is part of what marks the difference between modern cognitivist theories and associationism. He contends that the mind must employ a compositional syntax and semantics (that is, there must be syntactic principles for composing mental representations such that the semantic interpretation of a composed string is governed by the syntactic rules by which it is composed); otherwise crucial features of cognition such as productivity and

systematicity could not be explained (Fodor, 1987; Fodor & Pylyshyn, 1988). It should be noted that Fodor characterizes productivity and systematicity first as features of natural languages, and then applies them to the mind. *Productivity* refers to the fact that it is always possible to create new sentences in a language. Fodor argues that it is similarly always possible for a mind to think a new thought. *Systematicity* refers to the fact that for any expression that is part of a language there are others that are related to it in systematic ways that are necessarily also part of the language. Thus, if *The florist loved Mary* is a sentence of English, so necessarily is *Mary loved the florist*. Fodor contends that the same principle applies to thought: any mind that could think *the florist loved Mary* could also think *Mary loved the florist*.

What is noteworthy is that rather than using principles of the mind to explain human capacity in language, Fodor's approach has used language to explain thought. Unfortunately, this has the effect of making language even more mysterious for we cannot hope to explain it by decomposing it in terms of other simpler mental capacities. Since for Fodor this language-like representational system underlies language learning, linguistic capacity cannot be explained by learning; rather, it must be part of the person's native cognitive endowment. In itself this is not an insuperable problem. It might, for example, be possible to give an evolutionary explanation of how the language module came to be. Unfortunately, Fodor blocks this move as well by arguing that animals that demonstrate cognitive capacities must already have a language of thought. Moreover, Fodor does not offer a proposal as to how a process of variation and selective retention would have generated an internal language of thought. Finally, such a proposal seems seriously at odds with current theories of how the brains of other animals operate. Formal symbol manipulation is profoundly unlike the kinds of processes we observe elsewhere in the biological domain and its emergence in us appears mysterious (Churchland, 1986).

Given the problematic aspects of this approach, it is worth at least considering some alternatives. One way to open up alternatives is to consider again the path that led to this approach. I have stressed two elements: first, one starts with the most complicated form of the language use and makes that the basis for study; second, one localizes the capacity to use language in a particular system or subsystem. By focusing on the most highly developed form of language, we are led to the properties of languages that seem hardest to explain in terms of anything simpler. By attributing language use to a particular system (the mind) or subsystem (the language module), we are led to attribute to that system the very characteristics that distinguish the phenomenon itself. This makes the mind or the language module incredibly powerful and renders its operation mysterious. The suggested alternative, then, is to focus on simpler forms of language use and to consider how control of the use of language might be distributed, not localized. I have discussed the first strategy elsewhere (Bechtel, 1993a,b) and approaches to studying how less complex forms of language can be acquired by other species is explored by Rumbaugh (this volume). In this paper I will explore the second strategy by investigating whether it is possible to distribute the control of language in such a manner that one can more readily explain its development. I will then show that this can have the beneficial effect of reducing the resources we must attribute to the cognitive system in order to process language.

2. Distributing Control of Language

The motivation for localizing control of language use in the mind/brain is that it is human cognizers who comprehend and produce linguistic structures. How could they accomplish this if the control of language use were not internal to them? The alternative is to construe linguistic ability as an emergent product of the mind/brain and a certain kind of environment. Complex products often emerge from the interaction of two or more entities, none of which itself exhibits the requisite complexity to account fully for the phenomenon.

A clear example of how interaction can produce an emergent product out of simpler components is found in the work of Herbert Simon. Simon (1980) invites us to consider the path of an ant as it traverses an uneven terrain on its way to its goal. The path might appear very complex. But the ant does not have to represent this complexity. All the ant must do is embody relatively simple procedures for detecting and following the most flat course that is roughly in the direction of its goal. The complex trajectory is the product of the ant's relatively simple procedure for deciding on a course of motion, and a structured environment.

In the case of language, there are two environments external to the cognitive system that are pertinent. One is provided by the physical symbols (sound patterns, manual signs, written characters) used in language. These physical symbols afford certain sorts of use (e.g., referring to objects) and composition (e.g., linear concatenation either in time or space) and so make composed structures available to language users. The second is provided by other users of the language. The communal use of language serves to maintain a system of using particular symbols to refer to specific objects and of employing particular ways of putting linguistic symbols together to achieve certain ends.

I am not going to develop a comprehensive account of the manner in which both the physical symbols and the social context of the cognitive system interact in the development of language, since I want rather to explore the implications of this perspective for assumptions about what must go on in the head of the language user. But as preparation for my primary endeavor I will offer a speculative sketch of how external symbols and social contexts interact with the cognitive system. To see the importance of external symbols, consider first some rather high-level cognitive skills and how the use of written symbols supports those activities. Rumelhart, Smolensky, McClelland, & Hinton (1986) provide an example from arithmetic. For most people, multiplying two three-digit numbers is too complex a task to carry out in one's heads. To simplify the task, we make use of conventions for writing numbers on a page, as such:

343
822

This permits us to decompose the multiplication task into component tasks, each of which we are able to perform simply by knowing the multiplication tables. The procedure we were taught in school enables us to proceed in a stepwise manner. We begin with the problem 2 x 3, whose answer we have already memorized. As a result we write 6 directly beneath these two numbers:

343
822
6

The external representation of the problem then points us to the next step, multiplying 2 x 4. What we have learned is a routine for dealing with the problem in a step-by-step manner, where each step requires limited cognitive effort (remembering an already learned result). A problem that would be quite difficult if external symbols were not available is rendered much simpler with external symbols.

The main challenge in learning a task such as this is to learn to write the symbols in the canonical format and to proceed in the designated step-by-step manner. There are, of course, other ways in which the problem could be represented, and other procedures through which it could be solved. For example, we could encode the problem and the steps in the solution in the following manner:

343 x 822
686 + 6860 + 274400
281946

Using this representation, however, requires using the appropriate procedures for it, and this requires some relearning of basic skills.

I have spoken here of the performance of each step as involving remembering of an already learned result. But it could equally be described as a process of pattern recognition and completion. This characterization seems highly suited for other cognitive tasks such as evaluating formal arguments and developing proofs in formal logic. In *Connectionism and the Mind*, Abrahamsen and I discuss the problems of teaching students to use the argument forms of formal logic (e.g., *modus ponens)*, and we argue that what students must learn is to recognize patterns in external symbols. Here the patterns are a bit more difficult for the patterns have slots for variables, and what is required to instantiate the pattern is that the symbols that fill the slots stand in the right relation to each other. Students who have difficulty distinguishing valid from invalid forms often have not determined what property a pattern must have to be an instance of a pattern type. For example, they fail to appreciate that the same filler must fill both the slots for *A* in the following argument in order to have an instance of *modus ponens*:

If *A*, then *B*
A
∴*B*

Once they recognize this and thus have mastered the patterns of various valid and invalid arguments, they are able both to evaluate arguments and to construct arguments of their own. Constructing proofs, we contend, is an extension of this ability. Now, in addition to recognizing and completing valid argument forms, students must learn the patterns that specify when steps of particular kinds are fruitful in order to derive the desired conclusion.

What I want to emphasize here is the crucial role external symbols seem to play in both arithmetic and logic. As we are learning skills such as those of logic we seem to need to have the symbolic structures externally represented. Students often require much practice to learn to distinguish basic valid and invalid logical forms. To teach these to students I have relied on computer aided instruction in which students confront large numbers of simple arguments in English prose and have to determine their form and validity. Observing students performing exercises on the computer, I observe that they find it helpful to write out templates of each argument form and to compare explicitly the prose argument to each of their templates. The cognitive demands of comparing two external symbolic structures seems to be much less than internally representing the symbols and performing the comparison. Even advanced symbol users often rely on external representations when the forms get complex. For example, it is much easier to apply the de Morgan laws to determine that *It is not the case that both the legislation will pass and the courts will not block it* is equivalent to *The legislation will not pass or the courts will block it* when the sentences are written on paper than when we have merely heard them and must perform the operation internally. When the comparison is yet more complex, we often find it useful to write the intermediates forms on paper. Pattern recognition, completion, and comparison seems to place relatively low demands upon our cognitive system in contrast with high level computations.

The challenge is to see whether, in fact, by use of external symbols we can perform the high level computations of logic and arithmetic using only pattern recognition, completion, and comparison abilities. In Bechtel & Abrahamsen (1991) I reported on the ability of a connectionist network to recognize and complete simple argument forms of sentential logic. Recently I have been demonstrated the ability of a connectionist network to construct simple derivations in sentential logic by successively writing new steps onto units of the input layer (Bechtel, in press). In the following sections I will describe connectionist simulations by others that suggest a similar approach might work in the case

of language. But first I need to sketch in a more theoretical manner how the framework advanced here might apply in the case of language.

As mature language users, we often think to ourselves linguistically. This reinforces the idea that our mental representations are language-like and that the rules for using language are natively encoded in our cognitive system. How could it be that we rely on external symbols in the case of language? One clue is found in the comparison of spoken and written language. Not only does our spoken language often deviate from syntactical norms, but generally we fail to notice these deviations when we listen to speech. However, when the same speech is transcribed, the deviations stand out clearly. Thus, precise conformity to principles of grammar seems much easier when we use external written symbols. Written words are, however, only one form of external symbol. Spoken words also constitute external symbols, albeit more transient ones. Spoken words persist momentarily as sounds, and with the aid of echoic memory humans are able to maintain a trace of those symbols over a period of a bit longer duration. These external symbols are available to us not only when we listen to others, but as we speak. As mature language users we may not rely greatly on feedback of the sounds we have uttered, but this feedback may be far more important to language learners. The child learning a first language must not only learn to utter the sounds of a language, but also to order the sounds to fit the established patterns used in that language. At first the child's insertion into the ongoing use of language may be a single sound or what to us is a single word. Even without the child having a specific intention in mind, the community may interpret this utterance, and so it may have consequences (Lock, 1980). Having learned that individual sounds can be used in communication, the child gradually learns the conventions or patterns for putting them together. What the child is learning is to generate and respond to patterns in external symbols.

Having suggested that linguistic symbols may be construed as symbols external to the language users, I want to stress two things. First, language use is first embodied in a social context. Eventually humans learn to use language privately as a tool for thought, but this is derivative of the public use of language. Much of the process of learning to use a language depends upon interacting in this social context in which the particular principles of language use of the community are exhibited. Moreover, there is incentive for the language learner to master the patterns of a particular language for only then can the individual learn from the sentences uttered by others and use language to gain his or her own objectives. Second, the external symbols of language (sounds, manual signs, lexigrams, written words) themselves permit a certain kind of composition. Sounds, for example, can be strung together sequentially and uttered with different intonations and modulations. Grammatical principles of word order and case endings are natural devices to apply to these kinds of entities. Manual signs provide additional dimensions for variation (e.g., place the sign is made), and these dimensions are employed for grammatical purposes in various sign language. The grammatical devices that are "chosen" by the linguistic community are exemplified in the linguistic strings that are employed in that community. What the language learner must do is learn to conform to these structures: to extract the meaning that is encoded in these structures and to produce strings of his or her own.

What are the implications of such an approach for the psychological explanation of language processing? What I would argue is that with a distributed conception of language we do not need to posit nearly as rich a structure of internal representations as has often been thought. In particular, we might not need to posit a syntactically structured representation of language in the head and to view language processing as the performance of computations upon this structure. Part of the strategy for reducing what needs to be posited within the language user is to envision the linguistic community, and not the cognitive system, as being the primary enforcer of principles of compositionality in the language and the external

medium in which language is encoded (sound patterns, hand movements, ink blots on a page) as being the locus in which composition is achieved. The cognitive system exists in a linguistically structured environment, and must conform to the demands of that environment. At least at the outset, the symbols it uses are the symbols of natural language, typically physical sounds. What the cognitive system must learn how to do is to use these symbols and put them together in appropriate ways. This requires recognizing and using patterns. I should emphasize that the task that remains for the cognitive system is not trivial. But it is a different task than is projected when the cognitive system is construed as have a native language-like representation system on which formal operations are performed.

3. Lowering the Requirements on a Mind that Can Process Language

Fodor and Pylyshyn's arguments for a syntactically structured internal representational system are directed against recent connectionist models of cognition. Connectionist networks consist of units or nodes which have activation values and are connected to each other by weighted connections. They operate by having units excite or inhibit each other as they pass their activations along the connections, thereby causing changes in the activations of other units (Figure 1). (For an introduction to connectionism, see Bechtel & Abrahamsen, 1991.) Fodor and Pylyshyn's chief complaint against connectionism is that it represents a return to associationism, and they contend that associationism has already been demonstrated to be inadequate to model cognition.

Insert Figure 1 about here

The reason to see connectionism as associationist is that the connections between units in networks constitute associative links between what is represented by these units. The central arguments against associationism stemmed from Chomsky, who evaluated the potential of various levels of automata to instantiate grammars and argued that automata operating on merely associationist principles lacked the computational power required for the grammars of natural languages. My reference to grammatical principles as patterns and to pattern recognition as the basic skill required to learn a language may seem to have been an attempt to reduce grammars to associative principles and thus to run folly of Chomsky's arguments. But connectionism and the program for accounting for language I am proposing here are not so easily undermined. First, I have been emphasizing external symbols and suggesting that what the cognitive system must do is to learn to use these external symbols. The external symbols provide the cognitive system with increased computational power. Using the model of a Turing machine, we might see the cognitive system as comparable to the read head of the turning machine. The read head is a finite state device, but obtains its much greater power by reading and writing symbols on a tape. For the cognitive system, the role of the tape is performed by the medium in the external world from which it can read symbols and to which it can write them. Thus, supplemented by a medium for external symbols, a connectionist system has capacities equivalent to a Turing machine. Second, a connectionist system with hidden units (Figure 2) is more than a simple association device. Hidden units are typically used to transform the input pattern into a different pattern from which the target output pattern can be generated. With sufficient hidden units, a multi-layer network can be trained to generate any designated output for any given input pattern, and is thus a powerful computational device.

Insert Figure 2 about here

However, while a network is of the same computational power as a Turing machine, connectionist models do not operate in the same way as Turing machines or symbolic

computers. It is the differences between connectionist systems and computers running traditional programs that has attracted many researchers to connectionism. For example, connectionist systems exhibit content-addressable memory and graceful degradation, and lend themselves to tasks requiring satisfaction of multiple soft constraints. Moreover, insofar as connectionist networks are neural-like in structure, they constitute an architecture that can more reasonably be thought to have evolved through evolution. My interest in using connectionism in this project, however, is not to defend connectionism *per se*. Rather, I invoke connectionist systems as exemplars of a class of dynamical systems in which we might model cognition. What is important for my purposes is that these systems differ from those that have classically been used to model cognitive performance in that they do not employ language-like internal representations and formal operations upon them. If such systems could, nonetheless, learn to use external linguistic symbols, they can help us lower the requirements on a mind that can process language.

While they do not use internal language-like representations, connectionist systems do employ representations. The patterns on input and output units are construed as representing information. Moreover, the patterns on hidden units serve representational roles (Hinton, 1986). Critics of connectionism such as Fodor and Pylyshyn have focused on these representations, and have argued that the reason connectionism must fail is that these representations are inadequate. The reason is that they are not built up according to compositional rules and so are not themselves syntactically structured in a manner that permits structure sensitive processing rules to be applied to them. The reason is that activation patterns in networks can only represent the presence or absence of features of objects or events, not relations between those features. Multiple units being on, for example, can indicate that multiple features are present, but cannot indicate whether the features are instantiated in one object, or in many. For example, units representing *red*, *blue*, *circle* and *square* are active in Figure 3, but from this one cannot tell whether the circle is being represented as red or blue, and similarly for the square. The consequence, according to Fodor and Pylyshyn, is that connectionist models will fail to exhibit productivity and systematicity, the two features that they had claimed all cognitive systems exhibit. By way of contrast, linguistic representations are structured. In particular, they employ compositional syntactic rules for composing strings of symbols, and the semantic interpretation of a string adheres to these principles.

Insert Figure 3 about here

Many connectionists have struggled with the question of how they should answer Fodor and Pylyshyn. In what follows I will examine two strategies connectionists are exploring, the first of which accepts the demand that mental representations employ a system of compositional structure, albeit not a system such as classical syntax, while the second departs more radically from that framework. My goal in reviewing these programs is to explore the potential for developing connectionist networks which, while not employing linguistically structured internal representations, nonetheless are able to learn to extract information from and encode information in external linguistic symbols.

4. Networks that Employ Functional Representations of Syntactical Structure

What distinguishes a classical linguistic representational system is that each of the components is explicitly designated by words in a sentence, and the relationship between the different entities mentioned is specified by the grammatical principles by which the sentence is structured. One alternative strategy connectionists have pursued has been to build connectionist systems in which compositional structure is preserved functionally, but not structurally (van Gelder, 1990). As with syntactic structures, the goal is to build up complex

structures, but not ones in which representations of the components entities can be identified in the compound representation. The goal is that one can recover the components and their relations from the compound pattern that is created. This will make it possible to keep straight, for example, whether it is the circle that is blue, or the square, and to perform computational operations upon these representations roughly comparable to those that can be performed on syntactically structured sentences.

One exemplar of this approach is Jordan Pollack's (1990) recursive auto-associative memory (RAAM). (Another exemplar is the use of the tensor product operation to build compound representations. These bind components of a representation into a compound from which they can later be extracted. See Smolensky, 1990; Dolan, 1991.) In addition to developing connectionist representations which would respect the order found in a symbolic representation, Pollack sought to develop representations of complex structure that could be of fixed length. The reason this is important is that the input layer of any given network is of fixed size, unlike a sentential representation which can grow in size as additional clauses are embedded or as propositions are linked by logical operators. A standard way to depict structured symbolic representations such as sentences, for which Pollack wants to construct compressed representations, is as a tree structure (Figure 4). For each word in a string or tree Pollack assigned a 16 bit activation pattern. The task for the RAAM is to develop a 16 bit activation pattern that represents the whole tree.

Insert Figure 4 about here

To accomplish this Pollack used the encoder network shown on the left in Figure 5. It has 48 units (3 sets of 16) on the input layer and 16 units on the output layer. The bit patterns for the words on the terminal nodes on the lowest branches of the tree (*Mary*, *loved*, and *John*) are supplied to the three sets of input units, and the pattern created on the output units representations the compressed representation of that branch. The process is repeated at the next higher branch. (The tree used in this discussion branches only to the right. However, if the tree also branched to the left or from the center, then the compressed representations for all the nodes with branches extending from them at a given level would first be formed, and these, plus any terminal nodes at the level, would then be supplied to form the compressed representation at the next higher level.) In this case, the patterns for *John*, *knew* and the compressed representation for *Mary loved John* are supplied to the input nodes for the second cycle. This is a recursive procedure, so it can be applied for as many branches as are found in a particular tree. The decoder network on the right in Figure 5 is then used to uncompress the representation. This involves supplying the compressed representation for the whole sentence to the input units; the uncompressed representations is then constructed on the output units. If the representation on the output units is not itself a terminal representation, it is again supplied to the input units and another uncompressed representation is constructed on the output units.

Insert Figure 5 about here

In order to obtain from the decoder network what was supplied to the encoder network, appropriate weights must be found for all of the connections. To train these weights, the two networks shown in Figure 5 are joined as in Figure 6, creating an autoassociative network. An autoassociative network is one that is trained to construct on its output units the very same pattern as is presented on its input units. As long as the hidden layer has fewer units than the input and output layers, but still enough to recreate the patterns employed on the input and output layers, then an autoassociative network can be used to

create compressed representations from which the whole can be recreated. The procedure for training the network is parallel to the one described above. One starts with the terminal nodes on the lowest branch, and supplies each of them to the input units. The network generates a pattern of activation on the output units. This is compared to the target output values (which are the same as the input values) and the difference (known as the *error*) is used to change weights through the network according to a procedure known as backpropagation (Rumelhart, Hinton, & Williams, 1986). This procedure uses a derivative of the error with respect to the activation values of the output units so as to change weights in such a way that the network is more likely to produce the target output when given the same input in the future. After applying this procedure at the lowest branches, one proceeds to the higher branches, using the compressed representation that was generated on the hidden units as the input for the appropriate node at the higher level. This actually is a rather complex procedure, since when the same tree is processed again in the future, the weights will have been changed, and the pattern created on the hidden units for the terminal nodes will be different. Hence, at the higher nodes a different pattern will be used as input and target output. Thus, during training the network is chasing a moving target. However, through repeated applications of this procedure the network is able to acquire weights that permit near perfect auto-association. The two parts of the network can then be detached and used in the manner indicated in the previous paragraph.

Insert Figure 6 about here

Pollack trained his RAAM on 14 sentences similar to the one shown in the tree in Figure 4. After training, the encoder network was able to develop compressed representations from which the decoder network could reconstruct all 14 sentences. The network's abilities were not limited to the sentences in its training set. Pollack tested the ability of the network to encode and decode correctly variations of sentences in the training set. For example, in the training set, four of the sentences of the form "X loved Y" were employed. Since four names were available in the lexicon the network used, sixteen such sentences were possible. When the network was tested on these, it was able to develop compressed representations from which it could regenerate the original sentence for all of them. Thus, the network's ability is not punctate, but seems to exhibit systematicity. On somewhat more complex sentences, the network made some errors. For example, when given the new input sentence *John thought Pat knew Mary loved John* the network returned *Pat thought John knew Mary loved John*, which had been one of the sentences in the training set. One might argue, however, that this sort of error is precisely the sort we expect from humans as well (for example, you might have had to go back to reread the sentence to notice the difference).

The first significant feature of the compressed representations formed by the RAAM is that they do not employ explicit compositional syntax and semantics. There is no obvious representation of *Mary* in the compressed presentation of *Mary loved John*. Yet, the network's capacity seems to be systematic to a significant degree. However, there is a second aspect to these compressed representations. It turns out that they can be used for other computational processes. Chalmers (1990), for example, used a similar RAAM to construct active and passive sentences and then trained an additional Transformation network to construct the compressed representation of the active sentence from a compressed representation of the passive sentence. Even when the Transformation network was trained on only a subset of the sentences on which the RAAM had been trained, it was able to generalize perfectly and create a compressed encoding of the passive sentence from which the RAAM decoder network could create the correct uncompressed representation. (The

performance was less impressive, achieving only 60% correct, on those sentences on which neither the RAAM nor the Transformation network had been trained.)

Blank, Meeden, and Marshall (1992) performed a variety of additional tests to exhibit the usefulness of compressed representations developed by RAAM networks. They employed a variation on the strategy used by Pollack and Chalmers. Their RAAM formed a compressed representation from two input patterns at a time and they encoded sentences by proceeding through them word by word. When the first word of the sentence was encoded, it was supplied to the left hand set of input units, and the right hand units were left blank. Subsequently, the compressed pattern created on the hidden units was supplied to the right hand input units, and the next word to the left hand input units. In one simulation they trained the network to encode 20 sentences each of the form *X chase Y* and *Y flee X* as well as 110 miscellaneous sentences. Then they trained a feedforward network to generate the compressed form of *Y flee X* from the compressed form of *X chase Y* using 16 of the compressed patterns. The network generalized perfectly to the four remaining cases, and handled correctly 3 out of 4 additional sentences of the form *X chase Y* that were not in the training set of the RAAM. (The one error consisted of the substitution of one word for another.) Blank et al. also demonstrated other operations that could be performed on compressed representations. For example, they used the compressed representations as inputs to networks that were trained to determine whether a particular feature was present in the encoded sentence (a noun of the *noun-aggressive* category or a combination of a noun of the *noun-aggressive* category and a noun of the *human* category). The network was 88% correct in detecting nouns of the *noun-aggressive* category on sentences on which the RAAM (but not the detector network) had been trained, and 85% on sentences on which neither network had been trained. The scores were nearly identical in the combination feature test.

While these demonstrations are limited and the degree of generalization is modest, they do suggest that one might be able to use functional representations of the grammatical structure of the sentence to perform operations that otherwise would seemingly require an explicit representation of the grammar. While Fodor and Pylyshyn contended that one required an explicit compositional syntax and semantics to model cognition, RAAM networks indicate that connectionists can develop and employ representations in which the compositional structure is only functionally present. The RAAM architecture offers a potentially important advance beyond classical modes of representation since the connectionist functional representations may have very useful properties. For example, the RAAM network may develop similar compressed representations of similar sentences. This is important since connectionist systems generally handle new cases by treating them in the same manner as similarly represented cases. This accounts for their ability to generalize. One result of this is that networks do not crash on new cases. A second is that when networks make errors, these errors generally are intelligible errors. For example, all but one of the errors Chalmer's network made when tested on sentences on which neither the RAAM nor the Transformation network had been trained involved substitution of one word for another in the same grammatical category.

In a sense, however, research with RAAM networks is still in the spirit of classical cognitive modeling that employed linguistic-like representations. The RAAM builds up a complete representation of the linguistic input on which operations can then be performed. While the representations do not exhibit explicit compositional structure and the operations performed on them are performed by connectionist networks, the representations nonetheless appear to play the same role as linguistically structured internal representations and the operations performed on them are comparable to ones performed by applying formal rules in classical systems. In the following section I will consider a far more radical approach, one

that does not involve an attempt to build up a complete representation of the syntactic structure of a sentence.

6. Doing Away with Internal Representations of Syntactical Structure.

The perspective I suggested in section 4 was that the cognitive system might be viewed as extracting information from externally encoded sentences and encoding information in them, but without developing an internal representation of the sentence. One way to pursue this is to train a network to perform a task that is not one of encoding the structure of the sentence but one of using the information presented in the sentence to perform another task. Both of the simulations that I discuss here employ what are known as *recurrent networks* (Elman, 1990). Recurrent networks are designed to take advantage of the fact that linguistic input, either from speech or writing, is usually sequential in nature. Yet, there are dependencies between different elements in the sequential input, with both the meanings and grammatical function of given words being affected by preceding and succeeding words. Standard feedforward networks are not able to accommodate this since, after the network processes a given input, it starts fresh with the next input. Thus, the whole input linguistic structure must be presented at once if the network is to utilize the dependencies between items. This has the disadvantage of letting the number of input units determine a maximum length of an input sentence. The solution employed in a recurrent network is to copy the activations on the hidden unit on a given cycle of processing back onto a special set of input units, designated *context units* (Figure 7). The activations on the context units thus provide a trace of processing on the previous cycle. Since the activation of hidden units on the previous cycle was itself partly determined by the context units whose activation values were copies of hidden unit activations on yet a previous cycle, the recurrent network can provide a trace of processing several cycles back.

Insert Figure 7 about here

The potential of recurrent networks to process sequential inputs such as occur in language is illustrated in a simulations by Elman in which a recurrent network was trained to predict as output the next item in a sequence. In one simulation the input was a corpus of 10,000 two- and three-word sentences employing a vocabulary of 29 words. The sentences were constructed to fit 15 different sentences templates, of which the following are two examples: NOUN-HUMAN VERB-INTRANSITIVE (e.g., *Woman thinks*) and (NOUN-HUMAN VERB-EAT NOUN FOOD (e.g., *Girl eats bread*). These sentences were concatenated, with no indication of the beginning or end of individual sentences, to form a corpus of 27,524 words, which were presented to the network one at a time. The network was trained to produce on its output units the next word in the sequence and after only six passes through the training sequence its outputs closely approximated the actual probabilities of the next words in the training corpus. Note that in the actual corpus used in training a given word could be followed by several different words, and so its predictions should reflect the frequency of successive words. This is what was found. Moreover, it is not enough for the network to attend simply to the current word. What follows a given word may depend on what proceeds it. For example, *woman eats* will be followed by either *sandwich*, *cookie*, or *bread*, whereas *dragon eats* can be followed by *man*, *woman*, *cat*, *mouse*, *dog*, *monster*, *lion*, *dragon*, as well as *sandwich*, *cookie*, or *bread*.

How did the network obtain this level of performance? The recurrent connections provided the hidden units with relevant information about what had preceded the current input. The statistical technique of cluster analysis provides a useful way of analyzing the information contained on the hidden units. This technique determines the similarities of the various patterns across the hidden units are determined and permits the generation of a

hierarchical tree structure displaying the similarity structure of the patterns. Elman found that the patterns on the hidden units were grouped into categories according to their grammatical function. Thus, nouns employed patterns on the hidden units that were more similar to other nouns than to verbs. Amongst nouns, those referring to animate objects formed one subclass, those referring to inanimate objects another. Amongst animates, non-human animals were distinct from humans, and aggressive animals were distinct from non-aggressive ones. Verbs were also categorized into groups, with intransitive verbs distinguished from transitive verbs for which a direct object is optional and from those where it is mandatory. What is interesting is that these are categories that the network learned to distinguish while performing a quite different task: predicting the next word in a sequence. Identifying grammatical categories was not a task that was explicitly taught to the network. Knowing how the sentences in the corpus were constructed, of course, we can see why these are distinctions it was useful for the network to make. It is also interesting to note that there is this much regularity in word sequences that a simple network could pick up on it. This network had no access to the meanings of any of the words. Elman cites Jay McClelland's characterization of this task as comparable to trying to learn a language by listening to a radio. Chomsky appealed to the poverty of the stimulus in linguistic input to argue that language learning was only possible with a native understanding of grammar, but the network has induced grammatical distinctions from a limited input (albeit generated from a quite simple grammar).

Elman's goal was simply to show that a recurrent network could become sensitive to temporal dependencies and used linguistic input to illustrate this. He was not trying to model a realistic language-processing task. In a more realistic language task, what the network should be trying to do is extract appropriate information from a structured sequence. The challenge is to see whether a network which does not develop an explicit representation of the sequence can accomplish this. A recent simulation by St. John and McClelland (1990) illustrates how this goal might be pursued. One way to interpret the processing of sentences is to construe it as a task of developing a conceptual representation of an event. From such a conceptual representation one can determine what thematic role the entities mentioned in the sentence are playing. Thematic roles are different than grammatical roles. The grammatical subject of a sentence might be the agent (e.g., *the cat chased the mouse*), patient (*the mouse was chased by the cat*), or instrument (*the rock broke the window*) of an activity. In their simulation, the available case roles are: agent, action, patient, instrument, co-agent, co-patient, location, adverb, and recipient. Sentences were input to the network one word at a time and the task for the network was to answer questions about what entity or activity filled a particular thematic role or what thematic role an entity or activity filled. Thus, the input to the network might be the sentence "The schoolgirl spread something with a knife." In response to queries, the network should output *schoolgirl* when queried as to agent, and *knife* when queried as to instrument. If queried with *spread* it should respond with action. In addition to specifying the actual filler, the network was also trained to respond with a number of features of the filler, such as *person*, *adult*, *child*, *male* or *female* for agents. Thus, when queried as to agent with the previous sentence the network should not only indicate *schoolgirl*, but also *person*, *child*, and *female*.

In their simulation, St. John and McClelland employed a rather complex network (Figure 8), which can be analyzed into two parts. The top part responds to the queries put to it on the probe units. The probe input will specify either a given thematic role or a given filler. This probe and units designated as the *sentence gestalt* feed into a layer of hidden units, which in turn generate a pattern on the output units which should specify both the thematic role and its filler. The key to the operation of the network is clearly the construction of the sentence gestalt by the lower part of the network. The inputs to this part of the

network are the current word of the sentence and the previous sentence gestalt, which represents a copy of the pattern constructed on the sentence gestalt units when the previous word was input. These are fed through a layer of hidden units to create a new sentence gestalt.

Insert Figure 8 about here

The whole network is trained by back-propagation so as to generate the correct answer to the probes. The training procedure required the network to generate responses to all the case role and filler probes for the whole sentence after each word was input. Thus, from the very first word of the sentence the network is required to guess all the role/filler combinations for the whole sentence. The psychological interpretation St. John and McClelland offer for this procedure is that the network is to be thought of as experiencing real world scenes which the sentences describe, and its task is to interpret the sentences in accord with the scenes. The functional significance of the training procedure, though, was to force the network to attend to the dependencies between words in its corpus so that it could predict what was likely to follow given words of a sentence. It is this training procedure which accounts for a significant part of the network's ability to develop semantic sensitivity.

St. John and McClelland trained the network on a corpus of over 22,000 sentences describing 120 different events. Multiple sentences can be constructed for each event since there are different words that can be used for the same entity or action (e.g., *someone*, *adult*, and *bus driver* can all be used to designate the bus driver), and not all components of the event must be mentioned in each sentence (e.g., if the bus driver is eating, the instrument may be included or omitted). The events are constructed from frames associated with the 14 verbs in the vocabulary (four of which could also be used in the passive). The procedures used to construct the events made some events far more likely than others, exposing the network to a number of rather sexist stereotypes. For example, the bus driver (always a male) is described as eating steak more frequently than soup, and is generally portrayed as eating with gusto, while the teacher (always a female) more commonly eats soup and does so daintily. 330,000 random sentence trials were presented to the network during training. The network learned to make correct thematic role/filler assignments to the active sentences more quickly, but at this point also began to make assignments for the passive sentences.

The network was able to process a wide variety of sentences. In some cases, such as "The schoolgirl stirred the kool-aid with a spoon," the semantics was sufficient to determine the thematic role/filler assignments. But in a passive sentences such as "The bus driver was given the rose by the teacher" the syntax is crucial. (To insure that the network was relying on syntactic information, it was trained with equal numbers of instances of the bus driver giving a rose to the teacher, and the teacher giving a rose to the bus driver.) The network also made correct thematic role/filler assignments in ambiguous sentences such as "The pitcher hit the bat with the bat", with sentences in which concepts were not explicitly instantiated such as "The schoolgirl spread something with a knife", and with sentences in which role fillers were not explicitly mentioned such as "The teacher ate the soup" (instrument not specified). One of the more interesting abilities the network exhibited was its ability to revise earlier assignments when information later in the sentence required it. In the sentence "The adult ate the steak with daintiness" the network must supply the individual for the general category *adult*. When only "The adult ate" has been input, the network assigns equal response values to bus driver and teacher as agents and to steak and soup as patients. But after the word "steak" is input, it judges bus driver and steak to be the two most likely fillers, since the bus driver is described far more often as eating steak. The network

at this point also supplies the filler *gusto* for the adverb role. Supplying "daintiness" as input, however, brings a reversal. Now not only does *daintiness* surpass *gusto* in activation strength, but *teacher* receives more activation than *bus driver*. While *steak* receives more activation that *soup*, *steak* declines in activation and *soup* increases. The network is thus able to update its interpretation of previous information as new information becomes available to it.

Because of limitations in the manner in which the output to queries was encoded, it was not possible to test this network's abilities to process complex grammatical constructions such as embedded clauses. However, in a further simulation St. John and McClelland demonstrated the ability of such networks to handle complex syntactical structures. In this case the network was trained on 56 different sentences using "give." The following are some examples:

The bus driver gave the rose to the teacher.
The bus driver gave the teacher the rose.
The teacher was given the rose by the bus driver.
The rose was given to the teacher by the bus driver.
The rose was given by the bus driver to the teacher.

The network was able to extract proper thematic roles and fillers from this corpus. Thus, the model seems to be well on the way to extracting information from the syntactical structure of English sentences.

What St. John and McClelland's network must do is extract a representation of an event from sentences whose words are presented sequentially. The classical way of approaching this task is to build a structured representation of the input and perform computations upon that to answer the queries. This is not, however, what this network does. It does construct an internal representation (the Current Sentence Gestalt) in the course of processing the sentence and it uses this representation to construct its output. But this representation is not a classical representation with combinatorial syntax and semantics. Moreover, it is not even a representation that is designed to be functionally equivalent to the whole. It is a representation that captures that information in the input that is relevant to the task on which it was trained.

This simulation demonstrates that, at least in limited cases, a network can extract information from syntactically structured representations without employing internal syntactically structured representations. This raises the prospect that humans too can comprehend sentences without representing the sentences in a syntactically structured internal code and performing formal operations upon it. Of course, this prospect may turn out to be illusory. St. John and McClelland's network can only process a small fragment of English and it remains a question whether networks of this kind could eventually handle the full range of complexity found in human natural languages. The answer to the question will only come from further empirical investigation. In conducting such investigations, however, we should be careful not to exaggerate human ability. In written prose we can retrace our steps as necessary when dealing with extremely complex sentences. In oral communication, however, we often make mistakes in comprehending complex sentences. Reviewing the pattern of external symbols is not something that a network of this design could perform and we should not expect the network to do better than people can with oral input.

Another limitation of this network is that it is only potentially capable of comprehending sentences. Can a design such as this work for production as well as comprehension? I can only speculate as to how linguistic production might be modeled by a network which does not employ an internal representation of the syntactic structure of the sentences it is producing. What one might do in a simulation is develop a network which uses for its input a sentence gestalt of the form used in St. John and McClelland's simulation and

train it to produce proper sentential outputs. The speech output network again might be a recurrent network with activations on a layer between the input and output units being recycled as input. However, the sentence gestalt, or some other semantic representation, might remain as a constant input during all the cycles of processing until a sentence is complete. On the output units the network would be trained to issue the words of a sentence in sequential order. In such a simulation, the connection weights would need to acquire the knowledge of how to produce grammatically correct speech, but there would be no internal grammatically structured representation.

It is not clear how good a performance a connectionist network could achieve on such a task. But we must again bear in mind what standards we should use in judging such a network. We should not expect it to produce the full range of sentences that linguists judge to be correct sentences of a language. Rather, we should only expect it to obtain levels found in actual human speech. Even this is a quite unrealistic expectation for a relatively simple, totally interconnected network. It is already apparent that more structured networks, in which, for example, modules perform different parts of an overall task, achieve better performances than vanilla feedforward networks (Jacobs, Jordan, Nowlan, & Hinton, 1991). Realistic performance on such language tasks will likely await new developments in network design. But the goal is clear: to teach a network to produce proper linguistically structured sentences without employing an internal representation of the linguistic product that is itself syntactically structured.

6. Final Reflections on Modeling the Internal Competencies Needed for Language

The goal of my discussion is to argue that the requirements on the cognitive system responsible for language might be significantly less than they have often been portrayed to be. Humans languages do have complex structures, such as those linguists have identified in the course of developing grammars for natural languages. But responsibility for comprehending and producing grammatical speech may not lie exclusively with the internal cognitive system. The system can utilize the resources of external symbols and can be constrained by social processes supporting and governing language use. As a result, fluent use of a language may not require internal representations of the grammatical structures upon which formal operations can be performed but only the ability to extract and encode meanings in such structures. That does not mean that the internal processes used by the cognitive system might not be quite complex. But they need not be of the same sort as linguists have developed for describing language. The cognitive system may only be part of the system responsible for language, and its internal organization may be quite different from that of the emergent product.

Abrahamsen (1987) argues that the tasks of linguistics and psycholinguistics are quite different. Linguists are analyzing language as a cultural product. Grammars such as Chomsky's provide representations of that product. But these grammars need not characterize the processing occurring when people comprehend or produce sentences of a language. Abrahamsen argued that psycholinguists should expect to have to reformat the grammars developed by linguists when they try to account for the psychological processes involved in language use. My proposal here is that psycholinguists may need to go a step further. The systems involved in processing languages may be dynamical systems that do not employ grammatically structured representations internally but which generate and comprehend external symbols. It is these external symbols that afford syntactic structuring and the communities of language users who must develop and regulate the use of these structures. Cognitive systems can interact with syntactically structured symbols by producing and comprehending them, but language is an emergent product of cognitive systems, external symbols, and communities of language users.

The simulations I have described here are only meant as demonstrations of how abilities to utilize grammatically structured linguistic items might be accounted for without internal syntactic representations. They are not meant to be serious models of language processing ready to be evaluated by human data. I am not even convinced that they represent the most fruitful way of exploring linguistic ability within a connectionist, or non-symbolic framework. They approach the task of language processing in isolation from other cognitive activities, and the needs of the organism to control its body in its environment. Thus, there is no real semantics for such models. A far more realistic approach might be to begin with a model of a system functioning in an environment, that is, a system with sensory capacities to absorb information from its environment and motor capacities to change its environment. Given such a system, the challenge would be to extend this capacity by making it possible for the system to extract information from linguistic symbols present as part of the environment or to produce linguistic symbols as a means to manipulating its environment. Such a system would presumably develop the capacity to use symbols semantically before it began to attend to the syntax of linguistic structures. Once the system had developed the processing ability to recognize the semantic import of linguistic structures, though, it might notice that the grammatical structure provided additional information and it might learn to respect the grammatical structure as it sought to extract information.

I will close with two qualifying comments. First, by arguing for reducing the demands on the internal processing system responsible for language I may seem to be also endorsing an empiricist view of language according to which linguistic ability is simply acquired using more generally applicable cognitive abilities. But my position is also compatible with a form of nativism (Bates & Elman, 1992, Bechtel & Abrahamsen, 1991). While I have been arguing for an approach which does not posit internal syntactically structured representations, I have not denied that the cognitive system that learns to comprehend and produce language may be highly structured and have processing capacities quite different than those involved in other cognitive abilities. Presumably the neural hardware underlying language processing had to evolve first in a context in which it was not used for language, and has only in recent evolution become specifically used for language processing (see Deacon, this volume). Nonetheless, the system might well have been preadapted to the demands of language processing to such a degree that language learning seems almost inevitable amongst humans. We know that even children who lack linguistic models begin to develop language systems as long as they have an appropriate medium for symbol development (Golden-Meadow & Feldman, 1977). This argues for a system predisposed to develop language, but not for a specific analysis of the internal nature of this system.

Second, I have emphasized the external symbols such as sounds and inscriptions that figure in language use. We have, however, learned to use language internally. It is salient for my purposes, however, that the use of external symbols comes first in children's linguistic development and that using language privately in our thinking is a later development (Vygotsky, 1962). My expectation is that private linguistic thought will utilize many of the same computational resources as overt speech in much the way that visual imaging utilizes the same neural substrates as visual perception (Farah, 1988). Having learned to produce speech, we may have learned to go through all but the steps of overtly pronouncing words, and to have used this aborted production in much the same way as we learned to use external symbols. We might, for example, use this production capacity to create echoic memories of symbols. If this speculation is correct, then even in private thinking we are using symbols as if they were external, and are manipulating them in the same manner as we might manipulate truly external symbols such as inscriptions on a page. That is, we might try producing a symbol string, and then determine appropriate modifications of it. The symbols remain external to the cognitive system that is producing and recognizing them, and that

production and comprehension system might not need a syntactical representation of the syntactically structured output it produces and comprehends.

References

Abrahamsen, A. A. (1987). Bridging boundaries versus breaking boundaries: Psycholinguistics in perspective. *Synthese*, *72*, 355-388.

Bates, E. A. & Elman, J. L. (1992). Connectionism and the study of change. CRL Technical Report 9202, Center for Research in Language, University of California, San Diego.

Bechtel, W. (1993a). Knowing how to use language: Developing a rapprochement between two theoretical traditions. In H. Roitblat, L. Herman, and P. Nachtigall (Eds.), *Language and Communication: Comparative Perspectives*, pp. 65-83. Hillsdale, NJ: Lawrence Erlbaum.

Bechtel, W. (1993b). Decomposing intentionality: Perspectives on intentionality drawn from language research with two species of chimpanzees. *Biology and Philosophy, 8*, 1-32.

Bechtel, W. (in press). Natural deduction in connectionist systems. *Synthese*.

Bechtel, W. & Abrahamsen, A. A. (1991). *Connectionism and the mind: An introduction to parallel processing in networks.* Oxford: Basil Blackwell.

Bechtel, W. & Richardson, R. C. (1992). *Discovering complexity: Decomposition and Localization as strategies in scientific research.* Princeton, NJ: Princeton University Press.

Blank, D. S., Meeden, L. & Marshall, J. B. (1992). Exploring the symbolic/subsymbolic continuum: A case study of RAAM. In J. Dinsmore (ed.), *Closing the gap: Symbolism vs. connectionism*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Chalmers, D. J. (1990). Mapping part-whole hierarchies into connectionist networks. *Artificial Intelligence*, *46*, 47-75.

Chomsky, N. (1959). Review of *Verbal Behavior*, *Language*, *35*, 26-58.

Chomsky, N. (1965). *Aspects of a theory of syntax.* Cambridge, MA: MIT Press.

Chomsky, N. (1980). *Rules and representations*. New York: Columbia University Press.

Churchland, P. S. (1986). *Neurophilosophy: Toward a unified science of the mind-brain.* Cambridge, MA: MIT Press.

Dolan, C. P. (1989). Tensor manipulation networks: Connectionist and symbolic approaches to comprehension, learning, and planning. AI Lab Report, University of California, Los Angeles.

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, *14*, 179-212.

Farah, M. (1988). Is visual perception really visual? Overlooked evidence from neuropsychology. *Psychological Review*, *95*, 307-17.

Fodor, J. A. (1975). *The language of thought.* New York: Crowell.

Fodor, J. A. (1987). *Psychosemantics: The problem of meaning in the philosophy of mind*. Cambridge, MA: MIT Press.

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

Hinton, G. E. (1986). Learning distributed representations of concepts. *Proceedings of the Eighth Annual Conference of the Cognitive Science Society*, pp. 1-12. Hillsdale, NJ: Erlbaum.

Golden-Meadow, S. & Feldman, S. (1977). The development of language like communication system without a language model. *Science*, *197*, 401-3.

Jacobs, R. A., Jordan, M. I., Nowlan, S. J. & Hinton, G. E. (1991). Adaptive mixtures of local experts. *Neural Computation*, *3*, 79-87.

Lock, A. (1980). *The guided reinvention of language.* London: Academic.

Pollack, J. (1990). Recursive distributed representations. *Artificial Intelligence*, *46*, 77-105.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning internal representations by error propagation. In D. E. Rumelhart, J. L. McClelland, and the PDP Research Group, *Parallel distributed processing: Explorations in the microstructure of cognition, Vol. 1: Foundations*, pp. 318-62. Cambridge, MA: MIT Press.

Rumelhart, D. E., Smolensky, P., McClelland, J. L., & Hinton, G. E. (1986). Schemas and sequential thought processes in PDP models. In J. L. McClelland, D. E. Rumelhart, and the PDP Research Group, *Parallel distributed processing: Explorations in the microstructure of cognition, Vol. 2: Psychological and Biological Models*. Cambridge, MA: MIT Press

Simon, H. A. (1980). *The sciences of the artificial.* Cambridge: MIT Press.

Smolensky, P. (1990). Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial Intelligence*, *46*, 159-216.

Skinner, B. F. (1957). *Verbal behavior.* Englewood Cliffs, NJ: Prentice Hall.

St. John, M. F. & McClelland, J. L. (1990). Learning and applying contextual constraints in sentence comprehension. *Artificial Intelligence*, *46*, 217-257.

van Gelder, T. (1990). Compositionality: A connectionist variation on a classical theme. *Cognitive Science*, *14*, 355-384.

Vygotsky, L. S. (1962). *Language and thought.* Cambridge, MA: MIT Press.

Figure Legends

Figure 1. An illustration of processing in a connectionist network. The activation levels of the four units are shown beneath their labels. The weights on the three connections leading to Unit 4 are also shown. The Netinput to Unit 4 is determined by multiplying the activation of each feeding unit by the weight on the connection and summing across the three feeding units. The activation of Unit 4 is then determined according to the logistic activation function shown in the upper right.

Figure 2. A simple three-layer feedforward network. Each unit in the input layer is connected to every unit in the hidden layer, and each unit in the hidden layer is connected to every unit in the output layer. The hidden units serve to transform the input pattern into a new pattern from which the output pattern can be constructed.

Figure 3. An illustration of a problem facing connectionist representations. The units for *red*, *blue*, *square*, and *circle* are all active, but there is no way to indicate whether it is the circle or square that is red.

Figure 4. A tree representation of the sentence "Pat thought John knew Mary loved John."

Figure 5. Encoder and Decoder networks. The network on the left is an Encoder network; a representation of three components is supplied on the input units, and a compressed representation is generated on the output units. If the component compressed is only part of a larger structure, the compressed representation can be used as an input to the network on a subsequent cycle. The Decoder network is on the right. A compressed representation is supplied as input to this network and a decompressed representation is produced on the output units. If the output representation is still a compressed representation, it can again be used as an input.

Figure 6. Pollack's (1990) full RAAM network, consisting of the combination of the two networks shown in Figure 5. In this case the network is trained to reproduce on its output units the same pattern as is presented on the input units. The pattern on the hidden units is the compressed representation. If the pattern being compressed is itself part of a larger pattern, the compressed representation can then be used as part of the input and output pattern for the larger pattern.

Figure 7. Elman's (1990) recurrent network. Patterns generated on hidden units during one cycle of processing are copied onto the context units, and thus provide part of the input on the next cycle.

Figure 8. St. John and McClelland's (1990) network for determining thematic roles and fillers from sentences. The two square boxes indicate input units. The input on the Probe units specifies either the thematic role or the filler to be generated. The final output units are then to designate the combination of thematic role or filler. The sentence is input one word at a time on the Current Word input units. This pattern is processed through a set of hidden units to create a Current Sentence Gestalt. When a subsequent word is presented, the Current Sentence Gestalt is copied onto the Previous Sentence Gestalt and provides part of the input. This recurrent connection serves to provide the network a representation of the part of the sentence that has already been input.