Research Report AI-1993-02

A Connectionist Model of Language from Sensorimotor Preadaptation

F. James Eisenhart

Artificial Intelligence Programs
The University of Georgia
Athens, Georgia 30602-7415

F. JAMES EISENHART
A Connectionist Model of Language from Sensorimotor Preadaptation
(Under the direction of BRUCE K. BRITTON)

Some theorists claim that language is produced by a special rule-processing module located in the association areas of the human neocortex. However, anthropological, neural, and comparative evidence suggests that language is produced by general sensory and motor mechanisms that are common to all mammals. On this view, the prior evolution of advanced cognition preadapted general sensory and motor mechanisms for language. This thesis presents a connectionist language model that is consistent with this hypothesis. The model uses general sensory and motor mechanisms to understand and produce English sentences. By doing so, it demonstrates that it is not necessary to postulate an unprecedented new brain adaptation like a special rule-processing module in order to explain language.

Index Words: Cognitive Science, Connectionism, Language, Neural Networks, Neuroscience, Psychology, Sensorimotor Coordination

A CONNECTIONIST MODEL OF LANGUAGE FROM SENSORIMOTOR PREADAPTATION

by

F. JAMES EISENHART

B.S., Duke University, 1989

M.S., The University of Georgia, 1993

A Thesis Submitted to the Graduate Faculty of The University of Georgia in Partial Fulfillment of the Requirements for the Degree

MASTER OF SCIENCE

ATHENS, GEORGIA

1993

TABLE OF CONTENTS

DEDICATION

This thesis is dedicated to my parents, who have always supported me in whatever I have done.

ACKNOWLEDGMENTS

I would like to thank Dr. Bruce Britton for listening to my ideas, supporting me with his grants, and guiding my research here at UGA.

Chapter I. Introduction

1.1 Two Theories of Language

How does the human brain produce language? There are at least two views on this subject. The first comes from linguistics, rationalist philosophy, and artificial intelligence. It asserts that the brain produces language by processing the same sort of grammatical rules that linguists use to describe syntax. This is done by what Chomsky (1980) calls a "language organ". He claims that just as the heart is a separate blood-pumping module that operates on different principles from the lungs, the language organ is a separate cognitive module that operates on different principles from the rest of the brain. Thus, the way to understand how language works is not to compare it to other brain systems; it is to look at its observable behavior. The observable behavior of language is rule-processing, so it is reasonable to hypothesize that the language organ works by rule-processing as well.

A second, opposing view of language comes from biology, neuroscience, and anthropology. It stresses that hypothesized neural mechanisms for language should be consistent with what we already know about brain function and cognitive evolution. One of the things we know is that full-blown language is a recent phenomenon, not more than 200,000 years old (Lieberman 1984). Advanced cognition is much older because *Homo erectus* was already making stone tools, hunting in groups, using fire, and building shelters 1.5 million years ago (Donald 1991). This implies that the evolution of language may have depended on the prior existence of a rich cognitive structure. One way that this might have happened is if cognition made it possible for evolution to co-opt existing neural mechanisms to produce language. This sort of co-opting is called *preadaptation*: a structure or function that originally served one purpose is recruited for a new one. For example, the lungs evolved from the swim bladders of fish through a process of preadaptation. In the case of language, the likely preadapted neural mechanisms are those for sensory perception and motor action (Kimura 1979; Lieberman 1984). These are the most basic functions of any nervous system because an organism must be able to sense relevant changes in its environment and make appropriate motor responses in order to survive. They are related to language in that language includes both a sensory component (speech comprehension) and a motor component (speech production). This suggests that general sensory and motor mechanisms may have been preadapted for language by the evolution of advanced cognition. I will call this idea the *sensorimotor preadaptation hypothesis*.

1.2 A Connectionist Model

In this thesis, I present a connectionist language model that is consistent with the sensorimotor preadaptation hypothesis. My purpose is to show that basic connectionist principles of association can explain how general sensory and motor mechanisms could have evolved into language. This is an exercise in plausibility rather that proof. Given our present understanding of the brain, we cannot prove that any mechanistic theory of language is true; however, in chapter III, I do show that the sensorimotor preadaptation hypothesis is consistent with several lines of empirical evidence, while Chomsky's language organ hypothesis is not. Therefore, it is possible that a theory like the one modeled here will turn out to be correct.

The basic principle underlying the model is that language has two parts: a cognitive part and a sensorimotor part (see figure 1.1). The cognitive part is descended from the advanced cognitive apparatus that existed before language. It is assumed to be heavily dependent on vision, since visual areas occupy over half of the primate neocortex (Sereno 1991). The model assumes that the cognitive part exists, but it does not try to explain how it works. Since cognition probably involves most of the brain, it would be difficult to construct even a simplified account of it. Such an account will probably be necessary for understanding the semantic and pragmatic aspects of language, but, for understanding the contributions of sensory and motor mechanisms, it is not. Therefore, the model represents cognition abstractly by using thematic role/filler pairs like Agent=*cat* and Action=*meow* as stand-ins for the actual high-level representations used by the brain (Fillmore 1968). The idea is that Agent=*cat* corresponds to the brain having a neural picture of a cat, real or imagined, in some higher-order visual area.

The model uses these cognitive role/filler pairs to explain the sensorimotor part of language. The purpose of this part is to link cognition to a set of prelinguistic sensory and motor mechanisms. This linking function is performed by three neural networks. The first is a sensory network that translates language inputs into a high-level cognitive representation. Its takes a sequence of words as its input, and it produces a set of thematic role/filler pairs and surface features as its output. The second is a motor network that translates a high-level cognitive representation into language outputs. It takes a set of thematic role/filler pairs and surface features as its input, and it produces a sequence of motor actions representing words as its output. Finally, the third network is what Jordan and Rumelhart (1992) call a forward model. Its purpose is to predict the sensory consequences of motor actions. The input to the forward model is a set of motor actions from the output of the motor network, and its output is a prediction of the sensory consequences of those actions. As will be discussed below, these predictions are used in motor learning.

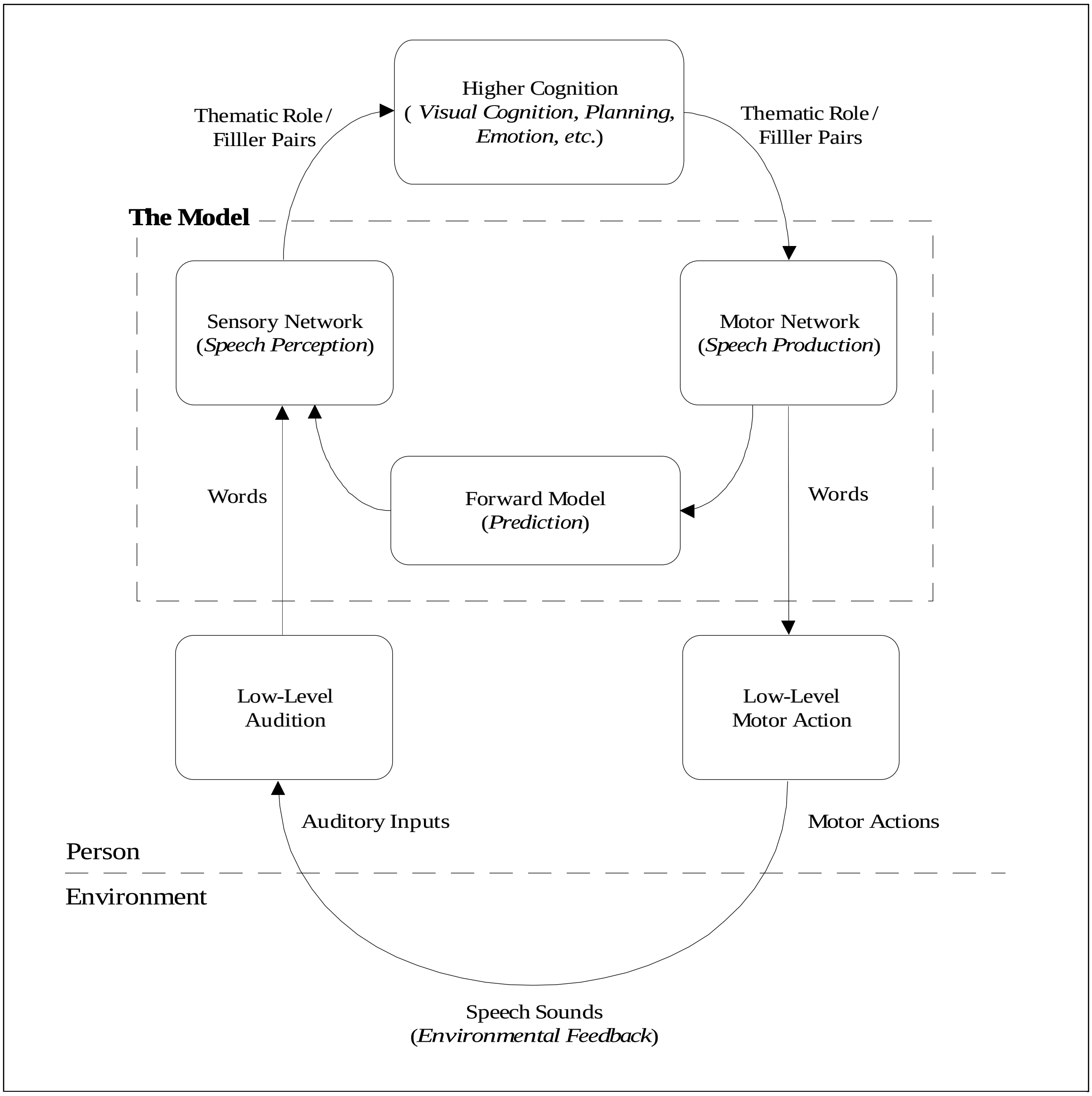

Figure 1.1. A schematic diagram of the model. The model includes three components (in dashed box): a sensory network, a motor network, and a forward model. The function of each is indicated in italics. Other components are hypothesized brain structures that are not explicitly modeled.

1.3 Organization of this Thesis

Chapter I has explained the motivation for the hypothesis that preadaptation of general sensory and motor mechanisms led to the evolution of language, and it has introduced the connectionist model that will be presented in this thesis. Chapter II discusses previous connectionist language models and their relationship to the present one. Chapter III presents empirical evidence supporting the sensorimotor preadaptation hypothesis and, by extension, the model. Chapter IV discusses general sensory and motor mechanisms and explains how they might produce language. Chapter V presents the model in more detail. It examines the architecture of the connectionist networks used, how they process sentences, and how they learn. Chapter VI explains how the model was trained and presents the training results. Finally, Chapter VII discusses what has been accomplished.

Chapter II. Related Work

There are generally two sorts of connectionist language models. Computer science models are concerned with achieving good performance on some difficult problem in natural language processing. The question that they ask is, How can a computer mimic the human capacity for language? They are successful if they achieve the desired computational result, even if they achieve it differently from the way that people do. On the other hand, psychological models attempt to explain some aspect of human language behavior. They ask, How can a neural network simulate the mental processes that people use to produce language? They are successful if they provide a convincing explanation for the observed behavior, even if the mechanisms that they use are computationally inefficient.

The dividing line between the two sorts of models is not absolute. Computer scientists have produced a number of models with interesting psychological properties, and psychologists have developed many of the computational techniques that are used in neural network research. However, the psychological models are closer in spirit to the one presented here, so this review will concentrate on them. Jain (1991) discusses the more computationally oriented work in his recent Ph.D. thesis, which is also an excellent example of the state of the art in connectionist parsing.

2.1 Spreading Activation Models

One class of language models is based on spreading activation. In these models, concepts are represented locally, by individual units, and relationships between concepts are represented by weighted links. Processing consists of several cycles during which the units pass activation back and forth. Typically, the model settles into a stable state that represents a coherent interpretation of the input conditions. This is a form of constraint satisfaction: the network units work in parallel to satisfy the set of constraints specified by the weights. These are soft constraints, meaning that they may not all be satisfied. Instead, the network may settle into a stable state that represents a compromise between two or more conflicting constraints. The strength of these models is that they offer an explanation for how people make judgments in the face of uncertain or even contradictory information. One weakness is that the constraints are usually set by hand, although there are automatic learning procedures for such models.[1]

One of the first models to apply spreading activation to language was Cottrell and Small's (1983) model of word sense disambiguation. This model is

[1] For instance, Boltzmann learning (Hinton and Sejnowski 1986).

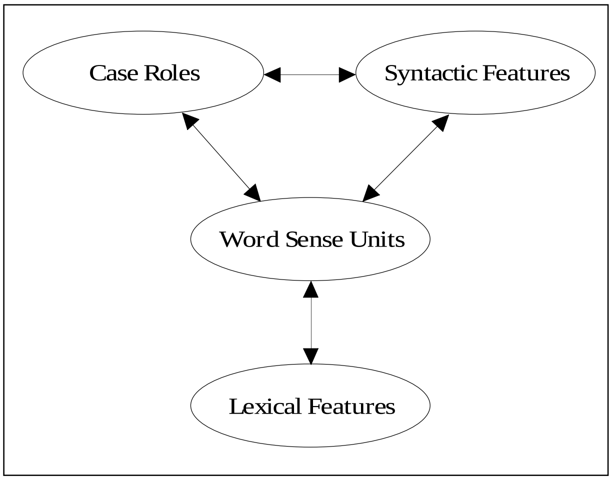

Figure 2.1. Cottrell and Small's (1983) model of word sense disambiguation.

based on the interactive activation architecture introduced in McClelland and Rumelhart's (1981) model of letter perception. As shown in figure 2.1, Cottrell and Small's model uses separate subnetworks to represent lexical features, word senses (i.e., baseball *bat* versus flying *bat*), case roles, and syntactic features. Connections within each subnetwork are mutually inhibitory, so alternative senses of the same word suppress each other. Connections between consistent units in different subnetworks are excitatory, so they reinforce each other. During processing, activation flows from the lexical units, through the word sense units, to the case role and syntactic units. An entire sentence is presented to the lexical level in parallel, and different word senses compete for activation by forming clusters of consistent units. The appropriate set of word senses usually wins this competition because it receives the most input from the lexical level and the most reinforcement from the case role and syntactic levels.

Waltz and Pollack (1985) used a similar network architecture to account for context effects in forming semantic interpretations. The principal difference between their model and Cottrell and Small's is that they use an additional set of context units to represent clusters of related words. For instance, the context unit for *hunting* is connected to the lexical units for *fire*, *deer*, and *bullet*. Their model uses these context units to produce different semantic interpretations for words depending on the surrounding context. For example, using the *hunting* unit, the model can decide that the word *bucks* refers to deer rather than dollars in the sentence *John shot some bucks*.

At a higher level, Kintsch's (1988; Kintsch and Welsch 1990) construction-integration (CI) model uses spreading activation to simulate the mental processes involved in reading and remembering a text. Kintsch has devised a procedure for automatically generating CI networks from texts. These networks consist of nodes representing the text propositions and links representing the relationships between the propositions. When Kintsch spreads activation through these networks, he finds that nodes representing propositions that people remember well become highly activated. Britton and Eisenhart (in press) have also found that a similar model captures some of the differences between mental representations generated by subject-matter experts and novices from reading a text.

2.2 Feedforward Models

Feedforward networks usually contain an input layer, an output layer, and one or more hidden layers. In the input and output layers, units represent particular concepts. In the hidden layers, units store intermediate computations. The weights connecting the layers are set using a supervised learning procedure like backpropagation (Rumelhart, Hinton, and Williams 1986). During processing, activation spreads forward from the input layer, through the hidden layers, to the output layer. One disadvantage of this architecture is that it has no memory; the activation values produced on one cycle cannot spread back through the network to influence those produced on a later cycle. In language models, this problem is typically overcome by using a time-sliced architecture in which several sets of input units are used to represent the input to the network at different points in time. However, this arrangement has the disadvantage that it arbitrarily limits the length of input sequence, so it is not an ideal solution.

One feedforward model is Rumelhart and McClelland's (1986) model of past tense learning for English verbs (see figure 2.2). They use a four-layer network that takes a phonological representation of the root form of a verb as its input and produces a phonological representation of the past tense as its output. The two interior layers of the network represent the root form and the past tense as a collection of context-sensitive phonemes called *Wickelphones* (after Wickelgren 1969). Each Wickelphone represents a phoneme, its predecessor, and its successor. For example, the *a* sound in *cat* is represented by the Wickelphone $_{k}\mathbf{a}_{t}$. The input and output layers are connected to their Wickelphone layers by a pair of fixed-weight coding networks, and the two Wickelphone layers are connected by an adjustable-weight associative network. During training, the root form is presented to the input layer at the same time that the past tense is presented to

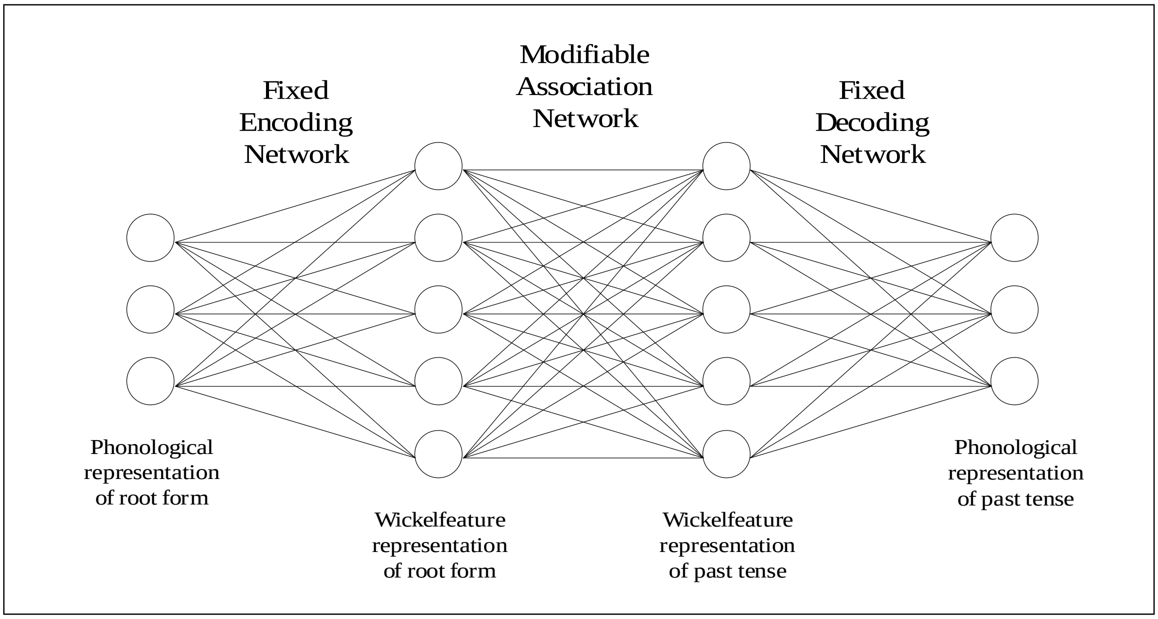

Figure 2.2. Rumelhart and McClelland's (1986) model of past tense acquisition.

the output layer. Both layers spread activation through their coding networks to their Wickelphone layers, and the perceptron convergence procedure (Rosenblatt 1962) is used to form an association between the two Wickelphone encodings. Later, when the input pattern is presented alone, this association allows the model to recall the output pattern. One interesting aspect of this model is that it learns past tense forms in the same three-stage order that children do. The model also generalizes regular forms to new verbs that it has not seen before, and it goes through a period when it overgeneralizes, applying its generalization rules too liberally, just as children do.

McClelland and Kawamoto (1986) used a similar two-layer model to translate the surface structure of a sentence into a set of case roles. Both layers represent words as a distributed set of multivalued semantic features like Form, Gender, Breakability, and so on. Each word has its own characteristic pattern of features. For example, the word *cheese* includes the features Form=*2-D*, Gender=*neuter*, and Breakability=*unbreakable*. The input and output layers encode these feature patterns with units that represent all possible pair of features. This is done so that the model can use combinations of features for learning without needing a hidden layer.[1] The feature pattern for each content word in a sentence is presented to the input layer, arranged by surface position. During training, a semantic training signal is presented to the output layer as a similar set of feature patterns, arranged by semantic case. The model then learns to associate the surface form with its semantic cases using the perceptron convergence procedure to change the weights. After training, the model is able to produce the contextually appropriate meaning of a word, choose the correct verb frame, fill in missing arguments with default values, and generalize to sentences with new words.

2.3 Recurrent Network Models

Unlike feedforward networks, recurrent networks have feedback connections that let them store contextual information. For language processing, this means that the previous words in a sentence can influence the interpretation of later ones. Consequently, recurrent networks have become popular in language processing. Most models are based on Jordan's (1986) recurrent network architecture (see figure 2.3) or a slight variant, the simple recurrent network (SRN) introduced by Elman (1990). The advantage of this architecture is that it uses fixed-strength feedback connections, so it can be trained with standard backpropagation. There are learning algorithms for recurrent networks with variable-strength feedback connections (e.g., Williams and Zipser 1989), but they tend to be computationally expensive. There are also more powerful recurrent architectures (see Mozer 1993 for a review), but they have not been used in language models.

---

[1] This was before backpropagation (Rumelhart, Hinton, and Williams 1986), so there was no known way to adjust the weights of a hidden layer.

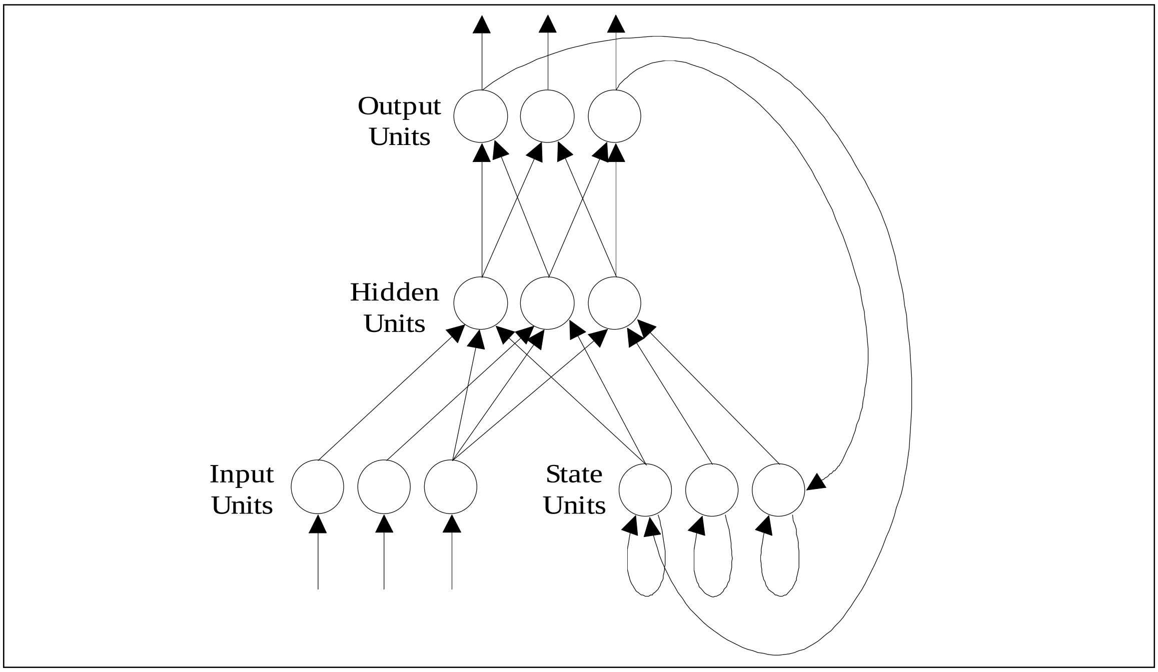

Figure 2.3. Jordan's (1986) recurrent network architecture. (Not all connections are shown.)

Elman (1990) showed that an SRN could learn to identify low-level constituents in a sequence of language inputs. In one experiment, he took 200 sentences and concatenated them together to produce a single list of 4,963 letters. He then used this list as input to an SRN that he trained to predict the next letter in the sequence. What he found was that the network predicted the first letter in a word poorly, but it did much better on subsequent letters. This shows that the network learned to use information implicit in the order of letters in words to learn the task. In a second experiment, Elman trained an SRN on a set of sentences concatenated together to produce a very long list of words. When he trained this network to predict the next word, he found that its output approximated the likelihood ratios for the possible next words. This shows that the network learned the semantic categories that Elman used to generate his sentences. Elman also confirmed this finding by performing a cluster analysis on the activations generated by the hidden units. This showed that words from similar semantic categories produced similar hidden unit activation vectors, indicating that the network had learned those categories.

St. John and McClelland (1990) used a two-part model to perform thematic role instantiation. The first part of their model forms what they call a *sentence gestalt*. This is just the output of a Jordan-style recurrent network, trained to produce a semantic representation of an input sentence. The sentence gestalt is then fed into second part of their model, a decoding network, which probes it for thematic role/filler information. The whole model is trained by presenting a sentence to the sentence gestalt network, and asking it a question with the decoding network. The error signal from the decoding network's answer is used to train the model by backpropagation. Recently, St. John has extended this approach to text comprehension (St. John 1992a) and language behavior during the performance of a task (St. John 1992b).

2.4 Relationship to the Present Model

The model presented in this thesis is like St. John and McClelland's (1990) in that it uses a recurrent network to map sentences onto a semantic role/filler representation. One difference between the two is that St. John and McClelland's model only uses the content words from a sentence as input. For example, it would abbreviate *the waitress hit the thief with the purse* as *waitress hit thief purse.* In the current model, the entire sentence is used as input.

Another difference is that, like all connectionist language models that I know of, St. John and McClelland's addresses the receptive component of language but not the productive component. Their model learns to understand sentences, but not to produce them. This seems to be part of a general tendency in psychology to emphasize the cognitive role of sensation and perception to a much greater extent than that of motor action, even though they seem to be two sides of the same coin. The current model simulates both language comprehension and production.

Chapter III. Empirical Evidence

The model presented here is based on a specific hypothesis, the sensorimotor preadaptation hypothesis, that predicts how language evolved and how it is produced in the brain. Accordingly, the empirical validity of this hypothesis is as important for the success of the model as the results that it produces. The empirical evidence for the sensorimotor preadaptation hypothesis falls into three categories:

1. Anthropological evidence which demonstrates that language evolved after advanced cognition.
2. Neurological evidence which shows that sensory and motor processing are the primary functions of the neocortex of all mammals, including humans.
3. Comparative evidence from the study of American Sign Language which shows that language can occur in sensorimotor modalities other than the usual auditory-vocal ones.

Each point is examined in more detail below.

3.1 Anthropological Evidence

If language evolved from the preadaptation of existing sensory and motor mechanisms, then hominids (members of the genus *Homo*, which includes humans) must have developed relatively advanced cognition before they developed language. Based on fossil, genetic and biochemical evidence, it is estimated that the hominid line diverged from the line that has led to chimpanzees 5 million years ago (Sarich 1980). In that time, hominid evolution has displayed two general trends. The first is towards increasing cultural sophistication. This is shown by archeological evidence for toolmaking, group hunting, the use of fire and shelters, and cave painting. The second trend is the development of a number of uniquely human anatomical specializations. The most obvious of these is an increase in brain size. Others include the development of erect posture, changes in hand anatomy, and the specialization of the vocal tract for speech. These changes did not occur continuously; they happened in abrupt jumps followed by long periods of gradual change. Based on these jumps, it is possible to construct a time-line for the major events in hominid evolution (see table 3.1).

The first major jump came with the australopithecines, about 4 million years ago. Their precise relationship to the hominid line is controversial, but it is

Table 3.1. Approximate time-line for hominid evolution, in years before present. After Donald (1991).

5 million years: Hominid line and chimpanzee split from a common ancestor

4 million years: Oldest known australopithecines
- erect posture
- shared food
- division of labor
- nuclear family structure
- larger number of children
- longer weaning period

2 million years: *Homo habilis*, oldest known hominids
- crude stone-cutting tools
- larger brain size

1.5 million years: *Homo erectus*
- much larger brain
- more elaborate tools
- migration out of Africa
- seasonal base camps
- use of fire, shelters

0.2 million years: *Homo sapiens*
- second major increase in brain size
- vocal tract assumes modern form
- first advanced cave paintings

0.05 million years: Fully modern humans

nearly certain that either the australopithecines were direct ancestors of the hominids or the two shared a common ancestor. In the latter case, the common ancestor probably possessed those traits which are common to both lines, so it is safe to assume that the hominid ancestors of that era had the adaptations found in australopithecines. The major change that occurred in the australopithecines was bipedalism. This meant adopting a less efficient means of locomotion and abandoning the safety of the trees. Australopithecines seem to have compensated by developing a more complex, cooperative social structure. They apparently did not compensate by being smarter. Their encephalization quotient (EQ), the ratio of their brain size to that of an average mammal of equivalent weight, remained within the range found in the great apes (Donald 1991), and they did not develop toolmaking. However, changes in australopithecine social organization probably set the stage for cognitive advances in later hominids.

The first major increase in brain size came with the earliest hominids, called *Homo habilis*, about 2 million years ago. Their EQ increased by about 30% over that of australopithecines, and they began to use primitive stone tools. However, *habilis* may not have been a true hominid species. Its members retained the overall build and facial features of the australopithecines, and the evidence concerning their EQ and tool-use is not certain. *Habilis* may really have been a transitional species between australopithecines and hominids.

The situation is much clearer with *Homo erectus*, who first appeared 1.5 million years ago. With *erectus* EQ increased by another 20%, and the physical appearance of hominids began to assume a much more human form. *Erectus* made significant cultural advances as well. One of these was the development of the Acheulian toolkit, which remained in use for over a million years and has been found in sites across Eurasia and Africa. Others were the use of fire, shelters, and seasonal base camps, and migrating long distances. Crucially however, *erectus* does not seem to have developed language. Based on anatomical reconstructions, Lieberman (1984) concluded that *erectus* lacked the elevated vocal tract that allows modern humans to produce speech; therefore, the rapid abstract communication that characterizes human language would have been impossible. Furthermore, the subsequent elevation of the vocal tract in *Homo sapiens* coincided with a cultural explosion that suggests the introduction of some important new social element (i.e. language), so it is likely that *erectus* did not have speech. As Donald (1991) suggests, *erectus'* own cultural advances may have been facilitated by some sort of prelinguistic communication system based on prosody.

The final jump in the evolution of the hominid line came with the emergence of *Homo sapiens* about 200,000 years ago. Brain volume increased by 20% to its present size, and the rate of cultural change began to accelerate continuously. Toolmaking improved gradually within the Mousterian culture that appeared about 100,000 years ago, then it was revolutionized in the Mesolithic and Neolithic cultures that followed. Language became possible when the vocal tract developed its modern form sometime between 100,000 and 200,000 years ago. Thus, it appears that language appeared contemporaneously with the final jump in hominid cognitive evolution.

What does all of this mean for the sensorimotor preadaptation hypothesis? The hypothesis claims that the prior existence of advanced cognition preadapted existing sensory and motor mechanisms for language. *Homo erectus* had over 80% of the brain volume of modern humans, an advanced toolmaking culture, and a cooperative social structure, yet he did not have language. So the time-line of hominid evolution is consistent with the sensorimotor preadaptation hypothesis. On the other hand, Chomsky's language organ hypothesis makes no prediction about the course of hominid evolution. Since it claims that language is an independent cognitive module, there is no reason why language had to evolve after cognition. It might just as well have been the australopithecines who developed language. Therefore, the language organ hypothesis fails to account for the evidence on hominid cognitive evolution.

3.2 Neurological Evidence

The cerebral cortex is the outermost part of the vertebrate brain. Those parts which are most developed in mammals are called the neocortex. This neocortex appears to be the site of all high-level cognitive functions, including language in humans. The neural circuits underlying these cognitive functions are not well understood, but some basic organizational principles of the neocortex are. Since the sensorimotor preadaptation hypothesis and Chomsky's language organ hypothesis claim that language is based on qualitatively different sorts of neural mechanisms, one way to evaluate them is to compare their hypothesized mechanisms with those suggested by the empirical results on neocortical organization.

One principle of neocortical organization is that the neocortex consists primarily of a number of unimodal sensory, motor, and limbic (motivational) areas. Primitive mammals have a small number of these areas, while carnivores and primates have more (Kaas 1987). The number of visual areas becomes especially large in primates. For example, monkeys have about 25 visual areas which consume over half of the neocortex (Felleman and Van Essen 1991). There are also a number of polymodal association areas with connections to multiple unimodal areas. At one time, it was thought advanced mammals had larger neocortices because they had larger association areas between their primary sensory and motor areas. It is now known that this is false. Carnivores and primates do have more cortex between their primary sensory and motor areas than primitive mammals like hedgehogs and rodents, but this extra cortex is taken up by larger and more numerous unimodal sensory and motor areas rather than larger association areas (Sereno 1991; Kaas 1987). Phylogenic trends, anatomical comparisons using fixed tissue, and PET[1] studies on live subjects suggest that this is true for humans as well (Sereno 1991). This is significant because theorists like Chomsky who claim that human cognition is produced by a number of uniquely-human brain modules traditionally assume that those modules are located in the association areas. If, as appears to be the case, human association areas are not proportionally larger than those of other primates, then it is not likely that they house a raft of new adaptations like Chomsky's language module. On the other hand, the sensorimotor preadaptation hypothesis is completely consistent with evidence that the neocortices of advanced mammals have grown by adding new sensory and motor areas.

A second principle of neocortical organization is that the neocortex has a basic local structure that is repeated in all areas. If it is laid out as a flat sheet and sliced vertically, a cross-section of the neocortex has six layers. These layers vary in thickness between sensory and motor areas, and each area has its own pattern of connections to other areas. However, the basic neocortical circuit, in terms of cell types and the pattern of connections between layers, is similar for all areas (Shepherd 1988). This is probably true because new cortical areas evolved through the duplication of genes for existing ones (Allman 1990). Subsequent functional modifications have almost certainly changed the computational properties of local circuits, but it is unlikely that they have produced an entirely new style of computation like grammatical rule-processing. This means that Chomsky's prediction that the brain contains independent cognitive modules

[1] Positron emission tomography (PET) is a technique for noninvasively monitoring activity levels in different parts of the brain.

that operate according to their own unique modes of processing is probably false. Most likely, all areas of the neocortex share a common mode of processing, which evolution has fine-tuned to perform particular functions, as the sensorimotor preadaptation hypothesis predicts.

3.3 Comparative Evidence

Language first evolved in the auditory and vocal modalities; however, they are not the only ones in which it can occur. We are all familiar with one example of this phenomenon: written language. The receptive component of written language is visual rather than auditory, and the productive component uses hand movements rather than movements of the vocal apparatus. But written language is not too different from spoken language. Written words have a one-to-one mapping onto spoken words, and both have a temporal order. This means that the same neural mechanisms might be responsible for both. Chomsky's language organ hypothesis can account for written language because it follows the same rules as spoken language. Only the sensorimotor transformations that connect the language organ to the world would have to be different.

However, sign language is a bigger problem. American Sign Language (ASL) is fully as complex as spoken language, but its structure reflects its visuospatial orientation (Bellugi, Poizner, and Klima 1989, 1990). For example, ASL has a well-defined syntax, but it is spatial rather than temporal. Signs are related to each other by their location in space rather than the order in which they are presented. Nominals can be associated with a particular spatial location, and modifiers applied to that location. This evidence is incompatible with theories like Chomsky's which postulate a special rule-based language processing center in the brain. Such a center would be specialized for temporal auditory-vocal rules, so it would be unable to handle the non-temporal syntax of ASL. On the other hand, if language is based on general sensory and motor mechanisms, then ASL's visuospatial orientation would naturally produce non-temporal rules.

Table 3.2. Comparing the predictions of the two hypotheses to the empirical evidence.

Line of Evidence	*Language Organ Hypothesis*	*Sensorimotor Preadaptation Hypothesis*	*Empirical Evidence*
Anthropological	Language is independent of cognition	Language depends on cognition	• Language evolved after cognition
Neurological	Rule-processing	General sensory and motor mechanisms	• More and larger sensory and motor areas • Small polymodal association areas • Common local circuit
Comparative	Auditory-vocal rules	General sensory and motor rules	• ASL uses visuospatial syntax

3.4 Evaluation

As shown in table 3.2, Chomsky's language organ hypothesis either makes the wrong prediction or no prediction at all for each of the three lines of evidence presented in this chapter, while the sensorimotor preadaptation hypothesis makes the right prediction for all of them. Of course, this does not prove that either the sensorimotor preadaptation hypothesis or the model presented in this thesis is right, but it does mean that they are worth considering.

Chapter IV. Sensory and Motor Mechanisms

At this point, I have presented evidence indicating that language is produced by general sensory and motor mechanisms, but I have not said what those are. Of course, there is a great deal that we don't know about the brain, so any discussion of brain mechanisms must remain somewhat speculative. However, based on what we do know, two mechanisms seem to be especially important in sensory and motor processing: topographic maps and associative learning. In this chapter, I explain (1) what each mechanism is, (2) why it seems to be important, and (3) what function it may perform. Then I suggest how topographic maps and associative learning might work together to produce language.

4.1 Topographic Maps

A topographic map is a systematically arranged collection of neurons with similar response properties. The best known example is the map of the retina in primary visual cortex. This map contains columns of cells that respond to bars of light that strike the retina at particular angles. These 'bar detectors' are arranged in groups, called hypercolumns, that each contain a full 360° worth of detectors for a single retinal location. In other words, when a bar of light strikes the retina at position (x,y) and angle θ , it triggers the θ -angle bar detector in the (x,y)-position hypercolumn. These hypercolumns constitute a topographic map because their arrangement in visual cortex preserves the neighborhood relationships between their receptive fields in the retina.

We know that topographic maps are important in sensory and motor processing because they have been found in nearly all sensory and motor areas (Kandel, Schwartz, and Jessell 1991). This includes both primary sensory and motor areas and higher-order ones. In primary areas, maps represent simple stimulus properties like the angle of a bar of light, the frequency of a sound, or the direction of force exerted by a particular muscle. In higher-order areas, maps represent more abstract stimulus properties like the form of an object, the relative pitch of a sound, or a complex pattern of muscle movements.

The function of topographic maps seems to be to organize sensory and motor information. In sensory topographic maps, each neuron combines information from several lower-level neurons. This produces a classification of the lower-level neurons' activity. For example, the bar detectors in primary visual cortex classify their incoming retinal signals by deciding whether they represent a bar of the appropriate angle or not. Connectionist networks can form similar topographic maps by a process like competitive learning (Rumelhart and Zipser

1985) in which groups of neurons compete for the right to represent a particular input pattern (von der Malsburg 1973; Linsker 1986). This forces nearby units to cooperate in order to win the competition; consequently, they learn to detect similar input features. Oja (1982) has shown that this process is equivalent to a principal component analysis of the lower-level activity patterns. The effect is that, if the input patterns have some regular spatial arrangement, the output units will form a sensory topographic map.

In motor topographic maps, each neuron sends information out to several lower-level neurons. When the map neuron is active, it instructs the lower-level neurons to execute a particular motor command. For example, neurons in a motor map in the supplementary motor area send signals to the lower-level premotor cortex that trigger complex movements like opening or closing a hand. Thus, it seems that, where sensory topographic maps form classifications of their lower-level input patterns, motor topographic maps generate lower-level elaborations of input patterns in the map.

How do motor areas learn to form these elaborative maps? This is not particularly clear. They cannot do it by statistically analyzing their input patterns, as sensory maps seem to do, because their input patterns—the map activations—cannot tell them which elaborations to produce. Instead, it seems that motor areas must somehow use sensory feedback from the environment to learn which lower-level motor actions should be produced by each map command. I discuss one way that they might do this in section 4.3.2.

4.2 Associative Learning

Associative learning enriches sensory and motor information by forming links between paired stimuli. Such links allows one stimulus to cue the recall of another. For example, the smell of freshly-baked bread might remind one of its taste.

At least two sorts of evidence suggest that associative learning is important in sensory and motor processing. First, behavioral evidence shows that we form associations all the time. The bread example suggests that this is true for sensory processing. In motor processing, familiar situations often trigger the recall of associated motor patterns. For instance, when we become distracted while driving a car, we all occasionally find ourselves following some familiar but unintended route.

Second, there is a cellular form of associative learning that appears to have an important role in sensory and motor processing. This mechanism, called associative long-term potentiation (LTP),[1] increases the strength of a synapse whenever its presynaptic and postsynaptic neurons are active at the same time. This is sometimes called Hebbian learning because, in 1949, the psychologist Donald Hebb predicted just such a mechanism would responsible for associative

[1] It is called associative LTP because there is also a non-associative version of LTP that is found, for example, in the CA3 region of the hippocampus. In the non-associative version, presynaptic input alone is sufficient to induce LTP; the postsynaptic cell need not already be active.

learning. LTP seems to be important for sensory and motor processing because it occurs in the hippocampus (Kandel, Schwartz, and Jessell 1991), a brain structure involved in the consolidation of episodic memories for storage in the sensory, motor, and association areas of the cerebral cortex. Furthermore, LTP probably occurs directly in cortical sensory and motor areas as well.

What function does associative learning serve? Primarily, it seems to give the brain several ways of accessing the same information. For example, most people can recall the taste of freshly-baked bread either by eating it or by smelling it. In language, this capability might allow people to recall a high-level visual representation of an object either by seeing it or by hearing a word that stands for it. This will be discussed in more detail below.

4.3 Sensory and Motor Mechanisms in Language

How do topographic maps and associative learning relate to language? Language has both a receptive component (comprehension) and a productive component (speech). According to the sensorimotor preadaptation hypothesis, the receptive component is implemented by sensory mechanisms that translate speech sounds into semantic events, and the productive component is implemented by motor mechanisms that translate semantic events back into speech sounds. If topographic maps and associative learning really are the fundamental sensory and motor mechanisms used by the brain, then they should be able to produce language. Below I discuss how this might be done.

4.3.1 Speech Comprehension

When a child learns to understand a language, he has at least two external sources of information available to him. First, he can hear sentences that proficient language users produce. Even if these are actually sentence fragments, as people tend to use in conversation, they give the child examples of what is legal in his language. Second, the child can observe the visual scene that accompanies each sentence. Since advanced cognition evolved before language, most of the cognitive apparatus that he needs to interpret this scene probably develops independent of language. If so, the child can use his interpretation of the scene to supply the meaning for any sentence that relates to it.

This suggests that a child's brain can learn to translate sentences into semantic events through a two stage process (see figure 4.1). First, it might form a topographic map that classifies the pattern of words contained in a sentence. This would have to be a *temporal* topographic map because a sentence is, unlike a bar of light striking the retina, extended in time rather than space. Temporal topographic maps do exist in nature, for instance in the auditory cortex of bats (Suga 1988) and songbirds (Margoliash 1983; Konishi 1985), so it is likely that the cortex can form such a map for sequences of words. In the figure, the topographic map has translated the sentence *the dog was bitten by the cat* into a set of surface features representing the phrase structure and voice of the sentence.

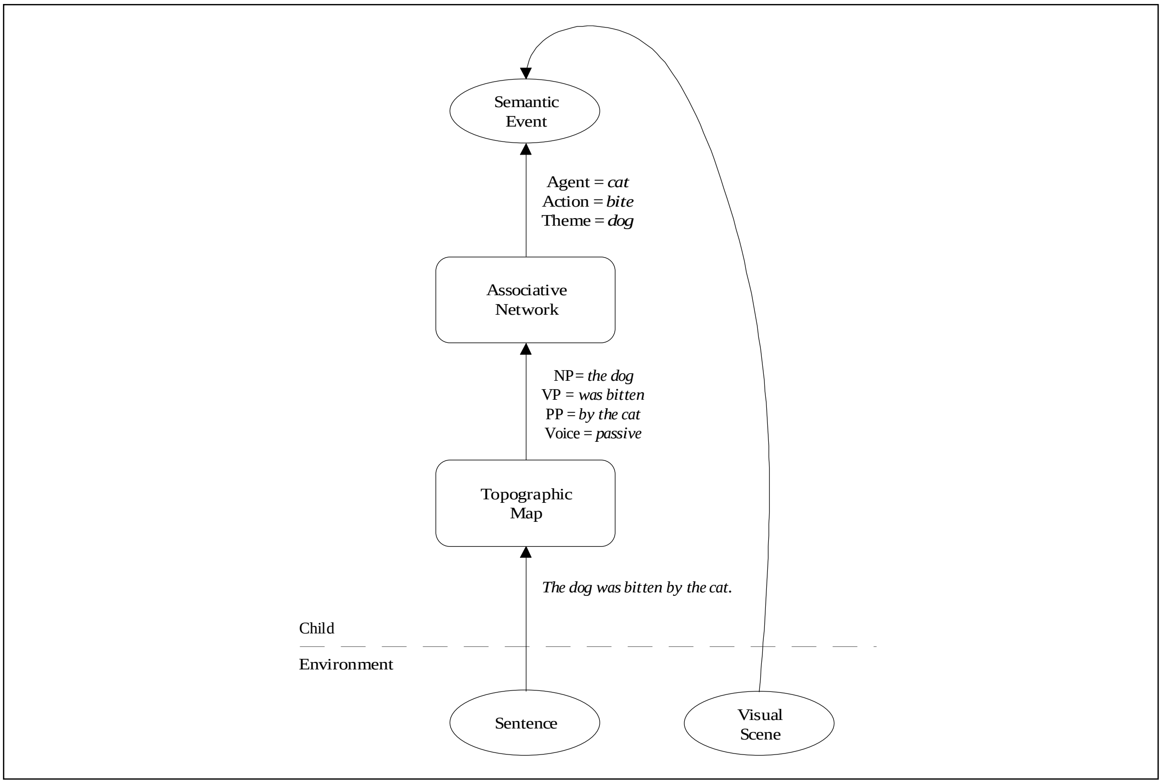

Figure 4.1. How the brain might use general sensory mechanisms to generate receptive language.

Second, the child's brain might use an associative network to link the active surface features in the topographic map to the active semantic features in the current visual scene. For example, in the figure, the associative network has linked the surface features for the sentence *the dog was bitten by the cat* to the semantic event Agent=*cat*, Action=*bite*, and Theme=*dog*. The associative network can learn to do this because both the sentence and the semantic event occur at the same time; therefore, a learning mechanism like LTP can form an association between them. However, there may also be some irrelevant semantic features present. For instance, the cat might have bitten the dog next to a chair, but the chair is irrelevant in the association process because it is not mentioned in the sentence. LTP probably learns to ignore irrelevant associations through a statistical averaging process that cancels them out over time.

4.3.2 Speech Production

When a child learns to produce speech, he has the same two sources of external information available to him as in speech comprehension: sentences that he hears and the corresponding semantic events. However, his learning task has changed. Now instead of matching up a sentence that he hears with a semantic event, he must learn to match up a semantic event with a sentence that he produces. At first glance, it seems the child's brain might do this through a two-stage process like the one used in speech comprehension (see figure 4.2).

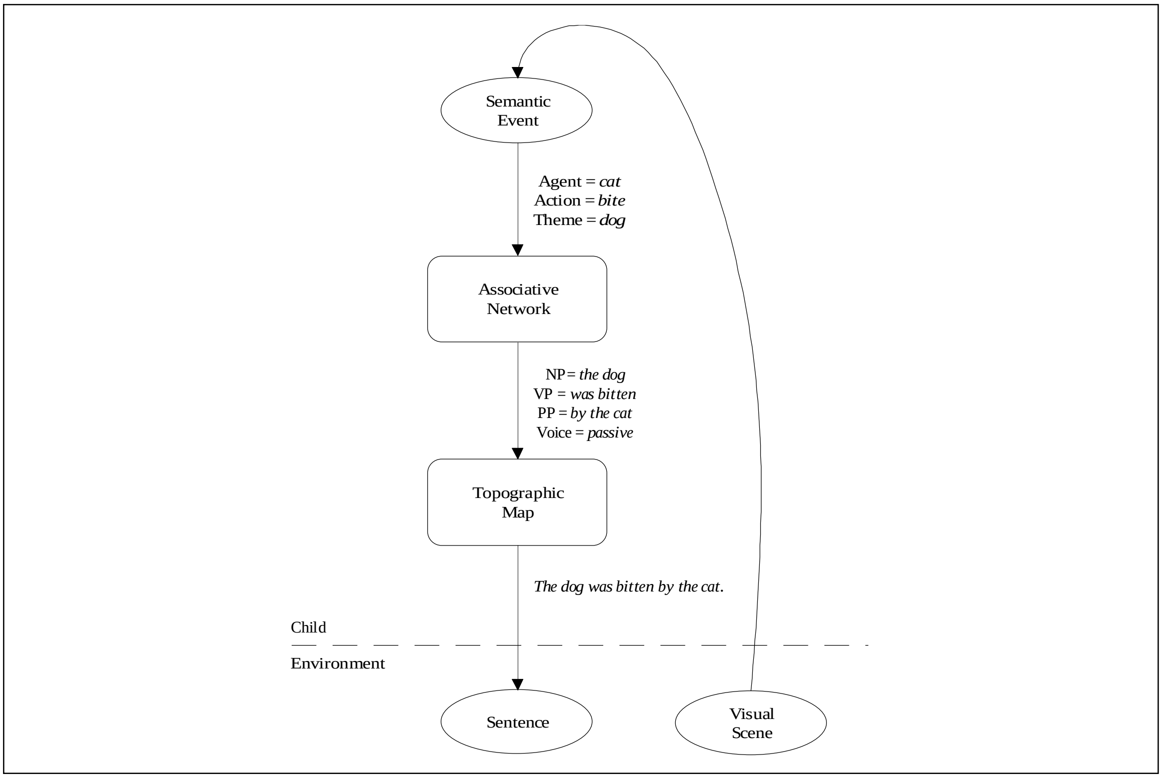

Figure 4.2. One way that the brain might use general motor mechanisms to generate productive language. This approach fails because there is no way to correct mistakes made by the topographic map.

First, an associative network would learn to link the active semantic features in the current visual scene to the active surface features in a motor topographic map. For example, in the figure the associative network has translated the semantic event Agent=*cat*, Action=*bite*, and Theme=*dog* into the surface features for the sentence *the dog was bitten by the cat*. Second, the motor topographic map would elaborate on these features to produce a sentence. In the figure, it has correctly produced the motor actions for *the dog was bitten by the cat*.

The problem with this scenario is that there is no way for the child's brain to form the motor topographic map. The associative network can begin the learning process by linking the active semantic features to some randomly selected set of surface features in the motor map, but there is no way for the motor map to learn which surface features should produce which words. It cannot learn through a statistical process, as sensory maps seem to do, because that would reinforce links between frequently paired surface features and motor patterns, even if the motor patterns were wrong.

One solution to this problem is to use environmental feedback to evaluate the sentences that the motor topographic map produces. This can be done by maintaining a *forward model* that links motor commands with their expected sensory consequences (Jordan and Rumelhart 1992). This forward model is an associative network that learns to predict the speech sounds that are produced by a set of motor actions (see figure 4.3). It is called a forward model because it

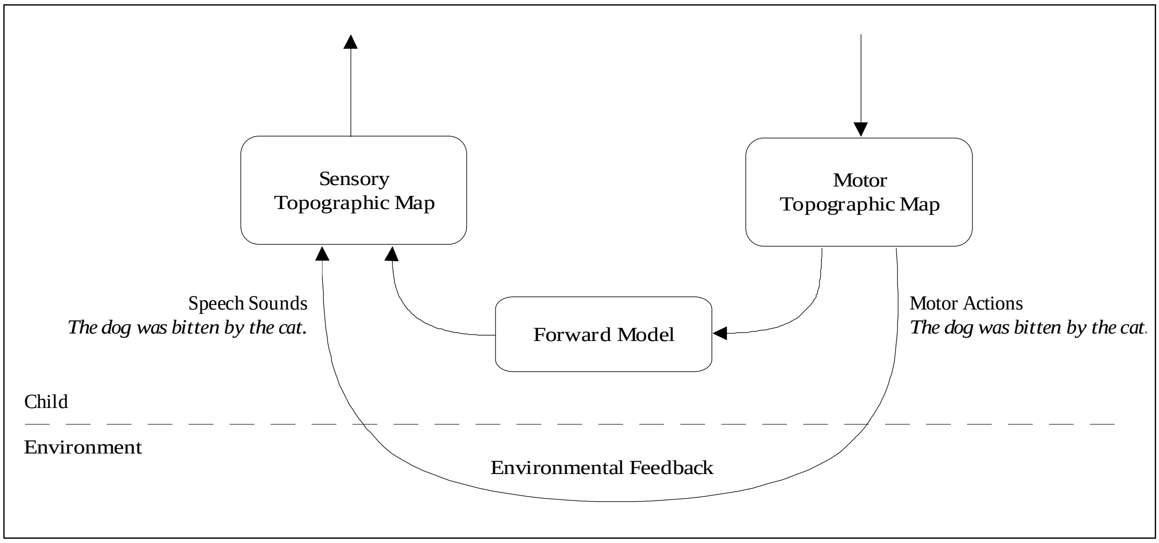

Figure 4.3. A forward model would predict the speech sounds generated by speech-producing motor actions.

predicts forward, from causes to effects, rather than backward, from effects to causes. An associative network can learn to perform this function because every time a child speaks, he can hear the sounds that he produces. Thus, the forward model can learn by matching up the motor commands that produced a speech act with the sounds generated by environmental feedback. In the figure, the child makes the motor actions for *the dog was bitten by the cat*, and he hears the speech sounds for the same sentence. By forming associations for a number of such examples, the forward model can learn to predict the speech sounds that will result from any speech-producing motor action.

If speech production uses a forward model, the motor topographic map can use sensory feedback from example sentences to learn to translate surface features into words (see figure 4.4). In this scheme, when a child's brain hears a sentence, it tries to silently imitate what it has heard. As before, the associative network links the semantic event supplied by the current visual scene to a set of surface features in the motor topographic map. The motor map then generates some set of motor actions that produce a sentence. Initially, these motor actions will be wrong. In the figure, the motor map has produced the sentence *the dog was kissed by the cat* when it was trying to imitate *the dog was bitten by the cat.* However, the motor map can now identify its error by feeding the motor actions that it produces through the forward model to generate a set of predicted speech sounds. It can then compare these predicted speech sounds to the actual ones in the example sentence. The difference between the two is an error signal that can be used to train the motor map. In the computer model presented in the next chapter, this is done by backpropagation.

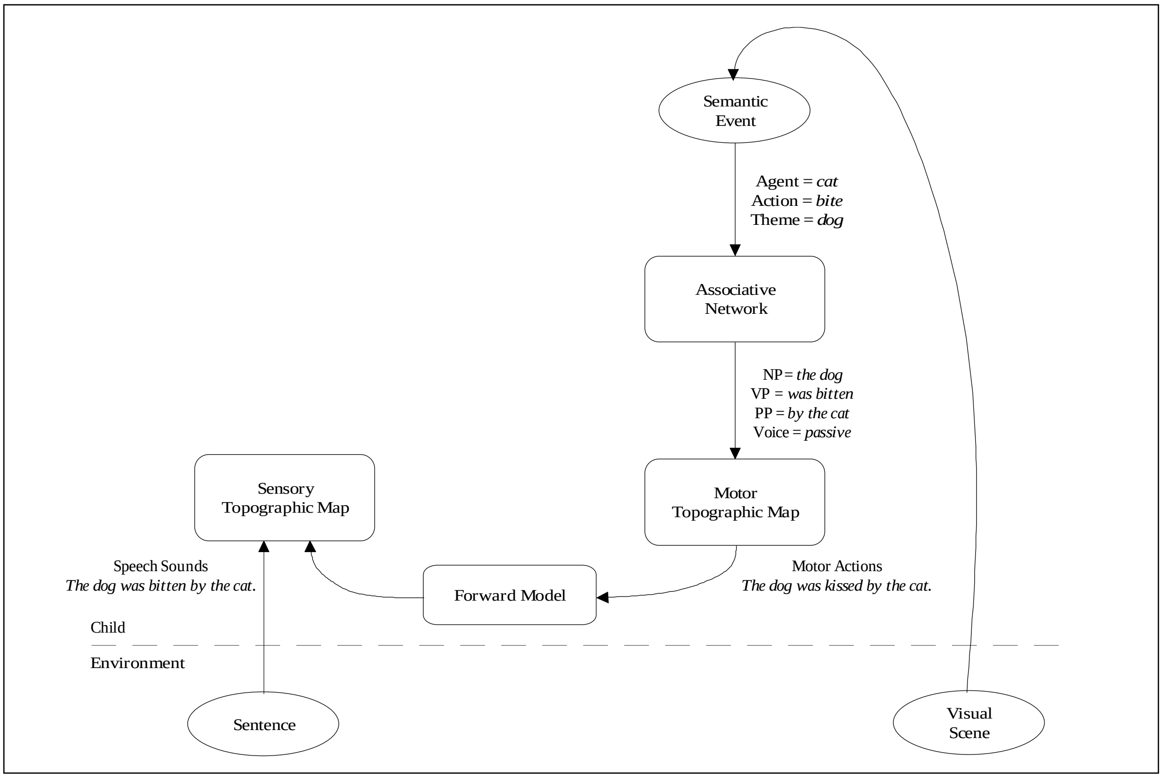

Figure 4.4. Another way that the brain might use general motor mechanisms to generate productive language. This approach works because the motor topographic map uses a forward model to compare its output to an example sentence heard by the sensory topographic map of the speech comprehension system. This way it can correct its errors.

Chapter V. The Model

5.1 Overview

In the computer model presented in this thesis, the speech comprehension and production systems are combined to produce a single language model. As shown in figure 5.1, this model has three components: a sensory network, a motor network, and a forward model.[1] The sensory network represents the speech comprehension system. It translates sentences into semantic events and surface features. For instance, it translates the sentence *the cat bit the dog* into the semantic event Agent=*cat*, Action=*bite*, and Theme=*dog* and the surface feature Voice=*active*. The motor network represents the speech production system. It translates in the opposite direction. Given a semantic event and a set of surface features, it produces the corresponding sentence. For the above example, it would produce the motor actions for *the cat bit the dog*. The forward model performs a function analogous to the one discussed for it in the last chapter; it links the sensory and motor networks by predicting the sensory consequences of a given motor action. For instance, when the motor network produces the action

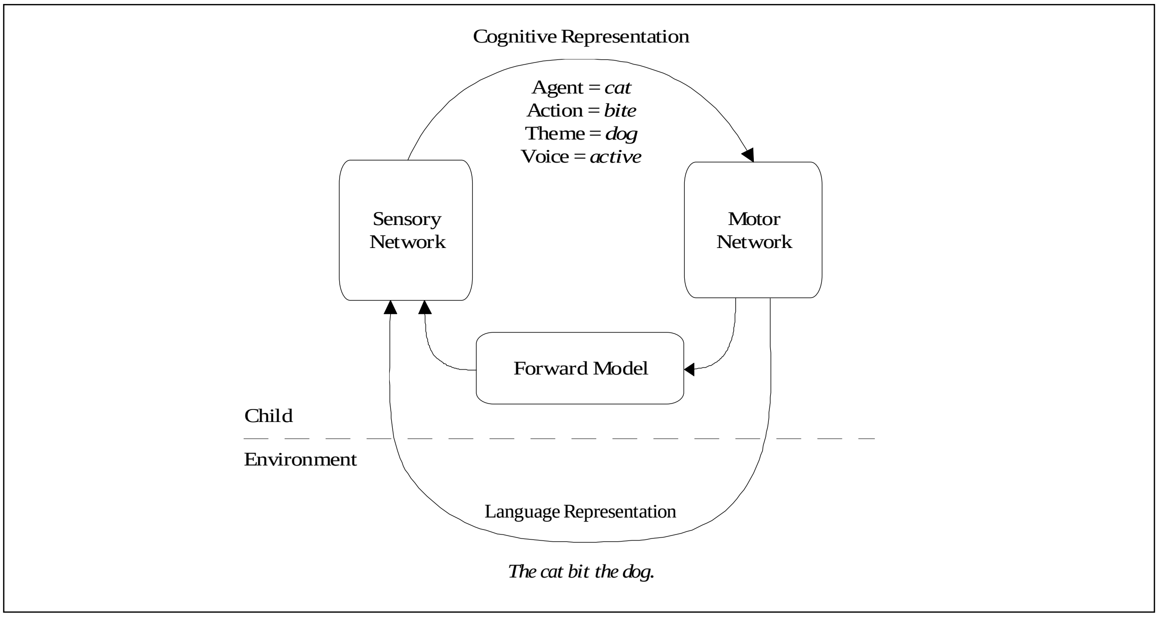

Figure 5.1. The model has three components: a sensory network, a motor network, and a forward model.

[1] I will always refer to the forward model by its full name to avoid confusion with the entire computer model that is presented in this thesis, but I will occasionally refer to the computer model as just "the model".

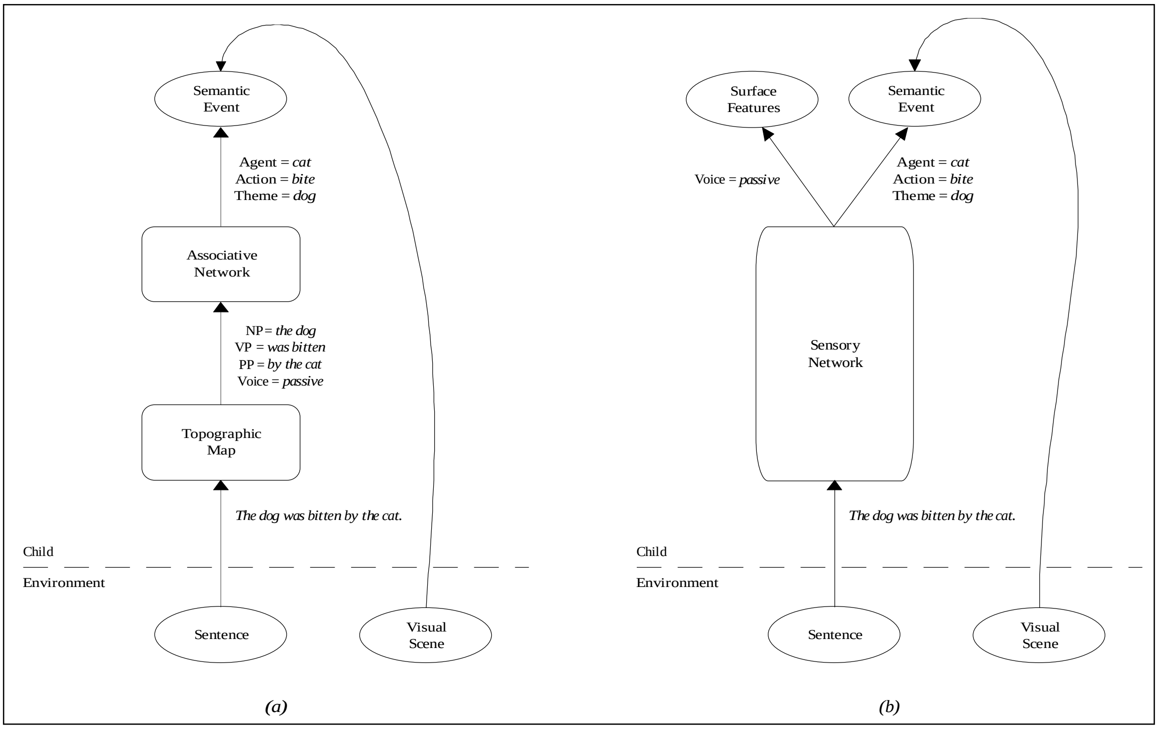

Figure 5.2. *(a)* The speech comprehension system that I suggest is used by the brain contains a topographic map and an associative network. *(b)* In the computer model, a single sensory network performs the functions of both.

for the word *cat*, the forward model predicts that the sensory consequences will be the *cat* sound.

5.2 Architecture

In the last chapter, I suggested that the brain systems responsible for speech comprehension and production each contain a topographic map and an associative network. In the computer model, these two networks are simulated by a single network that performs the same functions. For example, the suggested speech comprehension system from the brain and the sensory network from the computer model are illustrated in figure 5.2. The sensory network produces the same semantic event as the associative network of the brain system, in this case Agent=*cat*, Action=*bite*, and Theme=*dog*. It also produces whatever surface features are necessary to identify the surface structure of the sentence. In the brain, these features would be detected by the topographic map. In this example, the sensory network produces the surface feature/value pair Voice=*passive*, which distinguishes between *the cat bit the dog* and *the dog was bitten by the cat*.

Why use a single connectionist network rather than a separate topographic map and associative network? There are two reasons. First, existing connectionist networks that form topographic maps are not suitable for language processing. Connectionist paradigms like competitive learning (Rumelhart and Zipser 1985) can form spatial topographic maps like the one found in primary visual cortex, but they cannot form temporal topographic maps of the sort that might recognize patterns of words in a sentence. Also paradigms like competitive learning can only form sensory topographic maps, not motor maps, so they will not work for speech production.

Second, using single networks keeps the model simple. The model's purpose is to illustrate how advanced cognition might have preadapted general sensory and motor mechanisms for language, not to explain sensory and motor processing. Accordingly, this level of detail seems most appropriate.

5.2.1 Sensory Network

The sensory network uses is a modified version of Jordan's (1986, see figure 2.3) recurrent network architecture, shown in figure 5.3. Each unit in the input layer represents a word, and each unit in the output layer represents either a thematic role/filler pair or a surface feature/value pair. The network architecture differs from Jordan's in two ways. First, it uses separate state layers for the hidden and output units. This keeps the contextual information in the hidden layer separate from the semantic information in the output layer, helping the network learn more efficiently. Second, it adds a set of direct connections between the input and output layers. These help the network learn simple relationships between words and meanings quickly. For example, the word *cat* might mean Agent=*cat* or Theme=*cat*, but it cannot mean Action=*meow*. Using direct connections between the input unit for word *cat* and the output units for Agent=*cat* and Theme=*cat* cuts down on the number of word interpretations that the network has to consider. This frees up the hidden units to process contextual information.

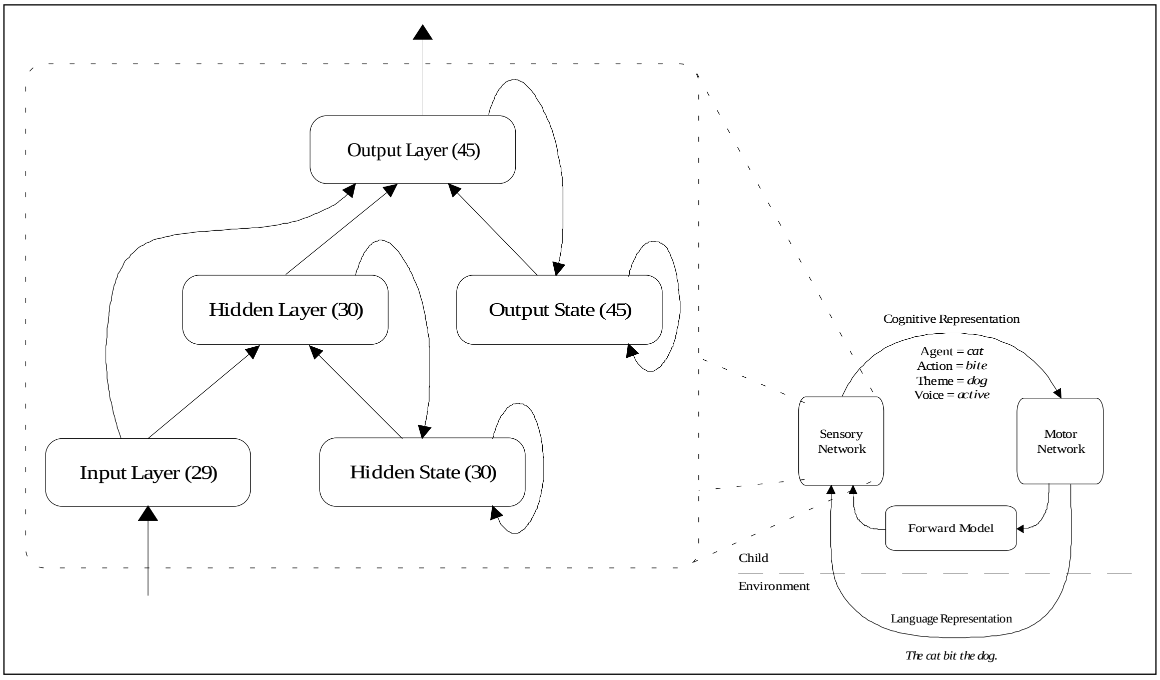

Figure 5.3. The sensory network uses a modified version of Jordan's (1986) recurrent network.

Table 5.1. Words used in the model.

Nouns		*Verbs*		*Prepositions*	*Miscellaneous*
billy club	policeman	barked	meowed	by	the
bone	purse	chased	saw	from	was
cat	ring	gave	seen	to	
dog	thief	given	taken	with	
fist	waitress	hit	took		
mouse		kissed	talked		

As shown in figure 5.3, the sensory network has 29 input units and 45 output units. The input units represent all of the possible words in a sentence (see table 5.1). There is one input unit for each possible word.

The output units represent all of the legal thematic role/filler and surface feature/value combinations (see table 5.2). There is one output unit for each thematic role/filler pair or surface feature/value pair shown in the table. The meaning of the individual thematic roles will be discussed in section 6.2. There are only three surface features because those are enough to distinguish between the alternative surface forms for any semantic event. The Voice surface feature refers to whether a sentence is active or passive voice (e.g. *the cat chased the dog* or *the dog was chased by the cat*). The Prep1 and Prep2 features identify the first and second prepositions in a sentence. For example, the sentence *the thief was hit by the waitress with the purse* would have Prep1=*by* and Prep2=*with*. In a sentence with only one prepositional phrase, Prep2 is not instantiated, and in a sentence with no prepositional phrases, neither Prep1 nor Prep2 is instantiated.

5.2.2 Motor Network

The motor network uses the same recurrent network architecture as the sensory network (see figure 5.4), but the roles of the input and output layers are reversed. In the motor network, the 45 input units represent the possible thematic role/filler and surface feature/value pairs, and the 29 output units represent the possible words in a sentence. The role/filler and surface

Table 5.2. Thematic role/filler and surface feature/value pairs used in the model.

Thematic Roles						**Surface Features**		
Action	*Agent*	*Theme*	*Source*	*Goal*	*Instrument*	*Voice*	*Prep1*	*Prep2*
bark	cat	bone	cat	cat	billy club	active	by	by
chase	dog	cat	dog	dog	fist	passive	from	from
give	policeman	dog	policeman	policeman	purse		to	to
hit	thief	mouse	thief	thief			with	with
kiss	waitress	policeman	waitress	waitress				
meow		ring						
see		thief						
take		waitress						
talk								

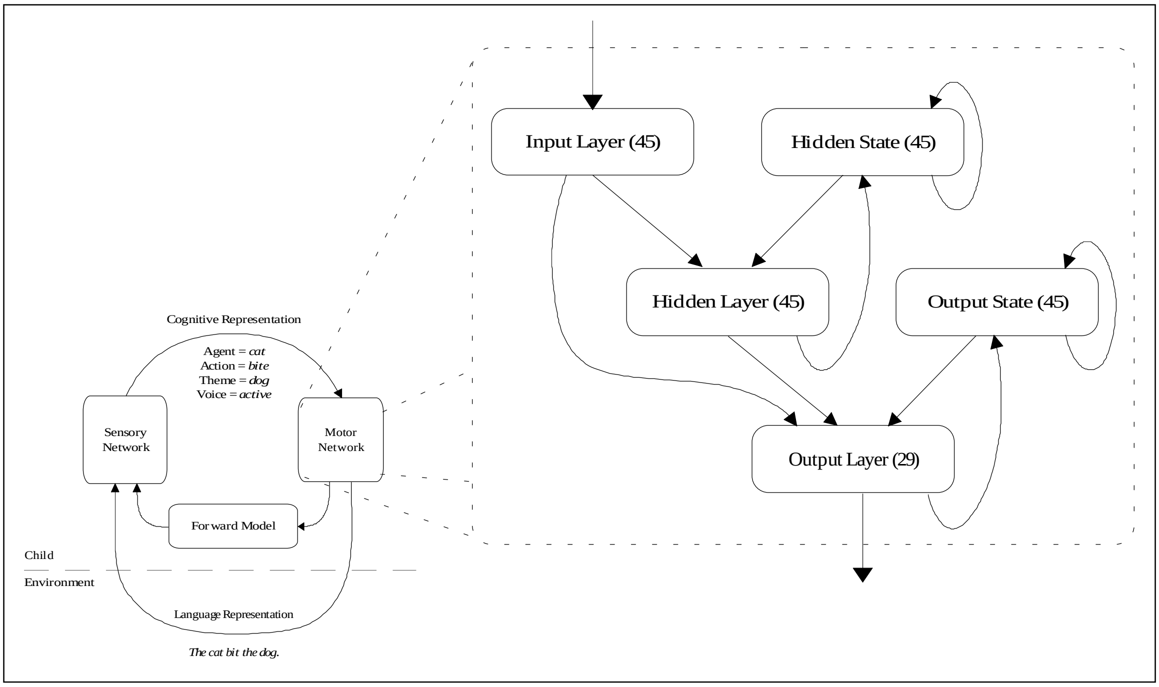

Figure 5.4. The motor network uses the same recurrent network architecture as the sensory network.

feature/value pairs and the words are the same as those for the sensory network (see tables 5.1 and 5.2).

5.2.3 Forward Model

The forward model predicts the sensory consequences (represented by the input to the sensory network) of a speech action (represented by the output of the motor network). Since the sensory inputs and motor outputs use the same word representation, the forward model only has to learn an identity mapping. A very simple network architecture can learn to perform this task. As shown in figure 5.5, the forward model uses a two-layer feedforward network. Each layer has 29 units. The input layer is just the output layer from the motor network. This means that every time the motor network produces a command to 'say' some word, the forward model uses the same command to predict the sensory consequences. The output layer of the forward model has the same unit layout as the input layer of the sensory network, but the two are physically separate. Consequently, the output of the forward model can be compared to the input of the sensory network to see if the forward model is making the right prediction.

5.3 Processing

Processing in both the sensory and motor networks involves a transformation between a temporal representation and a spatial one. The temporal representation is of the sequence of words in a sentence. In both the input layer of the sensory network and the output layer of the motor network, units represent individual words. A sentence is represented by a sequence of single

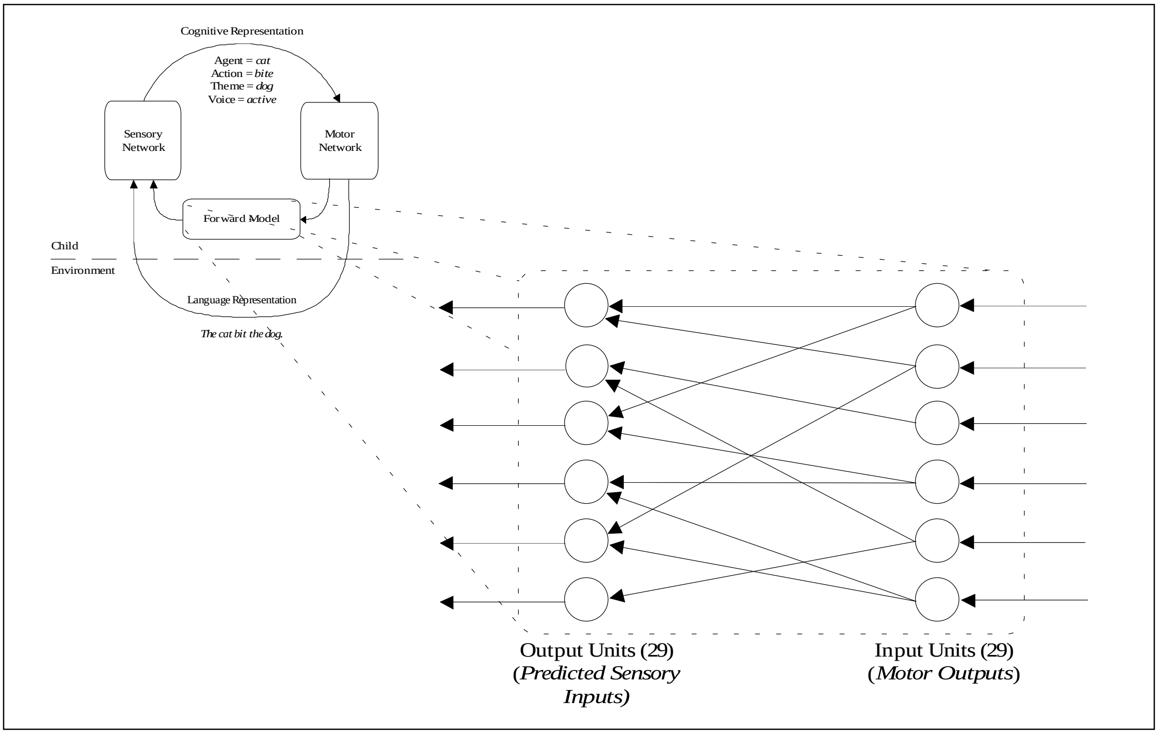

Figure 5.5. The forward model is a two-layer feedforward network. (Not all connections shown.)

word activations. For example, the sentence *the thief was kissed by the dog* is represented by the sequence of unit activations shown in table 5.3. Each row shows the unit activations for one time-step. On each time-step, only one word is active—the word whose position in the sentence matches the time-step number. For instance, the fourth word of the example sentence (*kissed*) is active at time 4. In both the sensory and motor networks, processing continues for as many time-steps as there are words in the sentence.

The spatial representation used by the sensory and motor networks is of the semantic event and surface features for a sentence. In both the output layer of the

Table 5.3. The temporal representation of the sentence *the thief was kissed by the dog.* (Not all word units are shown.)

	Word Unit										
Time	*the*	*cat*	*dog*	*thief*	*purse*	*ring*	*chased*	*hit*	*kissed*	*by*	*was*
1	**1.0**	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	0.0	**1.0**	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	**1.0**
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	**1.0**	0.0	0.0
5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	**1.0**	0.0
6	**1.0**	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7	0.0	0.0	**1.0**	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Table 5.4. The spatial representation of the thematic role / filler and surface feature / value pairs for the sentence *the thief was kissed by the dog.* (Not all units are shown.)

Agent			**Action**			**Theme**			**Voice**		**Prep1**	
cat	*dog*	*thief*	*bark*	*kiss*	*give*	*cat*	*dog*	*thief*	*active*	*passive*	*by*	*to*
0.0	**1.0**	0.0	0.0	**1.0**	0.0	0.0	0.0	**1.0**	0.0	**1.0**	**1.0**	0.0

sensory network and the input layer of the motor network, units represent individual thematic role / filler and surface feature / value pairs. A semantic event and a surface form for a sentence are represented by activating all of the appropriate units at the same time. This is a spatial representation because the pattern of activations is extended in space rather than time. For example, the semantic event and surface features for the sentence *the thief was kissed by the dog* are represented by the pattern of activations shown in table 5.4. The sensory network has to produce this spatial pattern from the temporal representation of the corresponding sentence, and the motor network has to use this spatial pattern to produce the temporal representation for the same sentence.

5.3.1 Sensory Network

Processing in the sensory network simulates what happens when a person hears and understands a sentence. The input to the sensory network is a time-varying pattern of activations representing a sentence, like the one shown in table 5.3. In response, the sensory network turns on each thematic role / filler and surface feature / value pair as its key word appears in the sentence. A key word is one that identifies a particular filler or value. As an example, consider the output for the sentence *the thief was kissed the by the dog,* shown in table 5.5. The key word for the Action role is always the verb from the sentence (in this case *kissed*), so Action=*kiss* is activated at time 4, when *kissed* appears in the sentence. The key word for the other thematic roles is always the noun from the corresponding noun phrase. For example, the key word for the Theme is *thief,* so

Table 5.5. The output of the sensory network for *the thief was kissed by the dog.* (Not all output units are shown.)

	Agent			**Action**			**Theme**			**Voice**	
Time	*cat*	*dog*	*thief*	*bark*	*kiss*	*give*	*cat*	*dog*	*thief*	*active*	*passive*
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	**0.8**	0.0	0.0	0.0	0.0	0.0	**0.3**	0.0	0.0
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	**1.0**	0.0	**1.0**
4	0.0	0.0	0.0	0.0	**1.0**	0.0	0.0	0.0	**1.0**	0.0	**1.0**
5	0.0	0.0	0.0	0.0	**1.0**	0.0	0.0	0.0	**1.0**	0.0	**1.0**
6	0.0	0.0	0.0	0.0	**1.0**	0.0	0.0	0.0	**1.0**	0.0	**1.0**
7	0.0	**1.0**	0.0	0.0	**1.0**	0.0	0.0	0.0	**1.0**	0.0	**1.0**

the network should activate Theme=*thief* at time 2. However, there is a slight problem here. The initial noun phrase in a sentence is ambiguous; *the thief* could also begin the sentence *the thief kissed the dog,* in which case we would have Agent=*thief* not Theme=*thief.* In situations like this, the network guesses which role to activate based on its past experience with similar sentences. Here, it guesses that *the thief* means Agent=*thief,* which is wrong. It sees this at time 3, when the word *was* tips it off that the sentence is passive voice. The network then corrects itself and activates Theme=*thief* instead of Agent=*thief.*

Processing continues for as many time-steps as there are words in the input sentence. On each time-step, activation flows through the layers of the recurrent network as shown by the arrows in figure 5.3. The hidden and output units are activated according to the standard logistic function

$$activation = \frac{1}{1+e^{-(netinput+bias)}}$$

where

$$netinput = \sum_i activation_i \cdot weight_i$$

and i ranges over all of the units connected to a particular hidden or output unit. At time t+1, each state unit is activated according to the function

$$activation(t+1) = activation(t) \cdot decay + reccurrent\_input(t)$$

where *decay* is a parameter between 0 and 1 that determines the rate at which the memory trace dies out, and *recurrent_input* is the activation of the state unit's hidden or output unit. The *decay* is 0.0 for the hidden state units and 0.6 for the output state units.

5.3.2 Motor Network

Processing in the motor network simulates what happens when a person has an idea and produces a sentence for it. The input to the motor network is a semantic event, representing the idea, and a set of surface features, representing the surface form that the person wants to use. Each thematic role / filler pair in the semantic event and surface feature / value pair in the surface form is clamped on at time 1. The network then runs for as many time-steps as it takes to complete the sentence. For example, the input to the motor network for the sentence *the thief was kissed by the dog* is shown in table 5.6. The semantic event is Agent=*dog,* Action=*kiss,* and Theme=*thief,* and the surface features are Voice=*passive* and Prep1=*by*. Those input units are clamped on for all 7 time-steps that it takes to produce the sentence, and all of the other input units are clamped off.

The output of the motor network is a sequence of motor actions that produces the sentence. For example, table 5.7 shows the output from the trained motor network for the sentence *the thief was kissed by the dog.* There is no problem with

Table 5.6. The input to the motor network for *the thief was kissed by the dog.* (Not all units are shown.)

	Agent			Action			Theme			Voice		Prep1	
Time	*cat*	*dog*	*thief*	*bark*	*kiss*	*give*	*cat*	*dog*	*thief*	*active*	*passive*	*by*	*with*
1	0.0	**1.0**	0.0	0.0	**1.0**	0.0	0.0	0.0	**1.0**	0.0	**1.0**	**1.0**	0.0
2	0.0	**1.0**	0.0	0.0	**1.0**	0.0	0.0	0.0	**1.0**	0.0	**1.0**	**1.0**	0.0
3	0.0	**1.0**	0.0	0.0	**1.0**	0.0	0.0	0.0	**1.0**	0.0	**1.0**	**1.0**	0.0
4	0.0	**1.0**	0.0	0.0	**1.0**	0.0	0.0	0.0	**1.0**	0.0	**1.0**	**1.0**	0.0
5	0.0	**1.0**	0.0	0.0	**1.0**	0.0	0.0	0.0	**1.0**	0.0	**1.0**	**1.0**	0.0
6	0.0	**1.0**	0.0	0.0	**1.0**	0.0	0.0	0.0	**1.0**	0.0	**1.0**	**1.0**	0.0
7	0.0	**1.0**	0.0	0.0	**1.0**	0.0	0.0	0.0	**1.0**	0.0	**1.0**	**1.0**	0.0

ambiguity here because all of the input units are clamped on or off at time 1. This means that the output unit for the correct word on each time-step is usually the only one activated. However, sometimes the network does not produce an activation of exactly 1.0 for the correct output unit. Here it has produced an activation of only 0.6 for *kissed* at time 4. This indicates that the network is still making some mistakes.

Processing follows the same rules as in the sensory network. Activation spreads out from the input units in the directions shown by the arrows in figure 5.4. Hidden and output units are activated according to the standard logistic function, and state units are activated using Jordan's recurrent activation function. The *decay* is 0.0 for both the hidden and output state units.

5.3.3 Forward Model

Processing in the forward model uses the motor actions from the speech production system to predict the sensory consequences heard by the speech comprehension system (see figure 5.5). The input to the forward model is a word

Table 5.7. The output of the trained motor network for *the thief was kissed by the dog.* (Not all output units are shown.)

	Output Unit										
Time	*the*	*cat*	*dog*	*thief*	*purse*	*ring*	*was*	*hit*	*kissed*	*by*	*was*
1	**1.0**	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	0.0	**1.0**	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	**1.0**
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	**0.6**	0.0	0.0
5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	**1.0**	0.0
6	**1.0**	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7	0.0	0.0	**1.0**	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

produced by the motor network. The forward model then activates its own output unit for the same word. This unit should have the same position in the output layer of the forward model as the corresponding word unit does in the input layer of the sensory network. For example, consider what happens on time-step 5 when a person says *the waitress gave the cat the mouse*. The motor network produces the word *cat*. This might be, leaving out some units, represented by the motor output vector <1,0,0,0,0>. If this motor command is executed and the person says the word, the sensory network will hear the word *cat* through feedback from the environment. This might be represented by the sensory input vector <0,1,0,0,0>. The forward model takes the motor output vector <1,0,0,0,0> as its input, and it produces the vector <0,1,0,0,0> as its output. In this way, it predicts the sensory consequences of a motor action.

During processing, the forward model spreads activation forward from its input units (the motor output units) to its output units. Its output units use the linear activation function

$$activation = \sum_i activation_i \cdot weight_i$$

where i ranges over all of its input units.

5.4 Learning

The computer model learns in two stages: a babbling stage and an imitation stage. The babbling stage is like the period like that children go through when they talk a lot but do not actually produce any words. The computer model uses this stage to train its sensory network and forward model. The sensory network is trained when the computer model hears a sentence. It learns by forming an association between this sentence and the semantic context in which it occurs. The forward model is trained when the computer model babbles. It learns by associating the motor actions that produced the babble with the sensory consequences that the babble generates.

The imitation stage begins once the sensory network and the forward model are trained. This stage is like the ones that children go through once they are able to form actual words. The motor network learns during this stage by imitating sentences that the computer model hears. The computer model does not actually say a sentence when the motor network imitates it; instead, the computer model feeds the motor output for the sentence through the forward model to predict what it would have sounded like. This prediction is then compared to the sentence that the sensory network actually heard, and the difference between the two is used to correct any mistakes made by the motor network.

5.4.1 Sensory Network

The sensory network uses standard backpropagation (Rumelhart, Hinton, and Williams 1986) to learn to translate sentences into semantic events and surface features. During training, each word in a sentence is presented to the

Table 5.8. The sequence of sensory network target vectors for the sentence *the policeman talked to the waitress.*

	Agent			**Action**		**Goal**		**Voice**		**Prep1**	
Time	*cat*	*thief*	*policeman*	*chase*	*talk*	*thief*	*waitress*	*active*	*passive*	*by*	*to*
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	**1.0**	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	0.0	0.0	**1.0**	0.0	**1.0**	0.0	0.0	**1.0**	0.0	0.0	0.0
4	0.0	0.0	**1.0**	0.0	**1.0**	0.0	0.0	**1.0**	0.0	0.0	**1.0**
5	0.0	0.0	**1.0**	0.0	**1.0**	0.0	0.0	**1.0**	0.0	0.0	**1.0**
6	0.0	0.0	**1.0**	0.0	**1.0**	0.0	**1.0**	**1.0**	0.0	0.0	**1.0**

network sequentially. On each time-step, the target vector includes those role / filler pairs and surface features whose key words have already appeared in the sentence. For example, the sequence of target vectors for the sentence *the policeman talked to the waitress* is shown in table 5.8. Only Agent=*policeman* is active at time 2, but Action=*talk* and Voice=*active* are also active at time 3.

On each time-step, the network is trained to produce the appropriate target vector. The error for each output unit is given by the function

$$error = (target - activation) \cdot activation \cdot (1 - activation),$$

and the error for each hidden unit is

$$error = activation \cdot (1 - activation) \cdot \sum_i error_i \cdot weight_i$$

where i ranges over all of the output units that the hidden unit connects to. At time t, each weight is changed according to the function

$$\Delta weight(t) = \varepsilon \cdot error \cdot activation + \alpha \cdot \Delta weight(t-1)$$

where ε is the learning rate and α is the momentum.[1] The network was trained with an ε of 0.2 and an α of 0.9. The training results are presented in chapter VI.

5.4.2 Forward Model

The forward model uses a modified form of backpropagation to learn to associate motor actions with their sensory consequences. On each learning trial, a babble is generated by assigning a random value between 0 and 1 to each motor output unit (see table 5.9). This babble is then converted into a set of sensory consequences. Since the motor output units and sensory input units use the same word representation, the sensory consequences are identical to the motor babble.

[1] The momentum term dampens oscillations in the weights. See Rumelhart, Hinton, and Williams (1986) for details.

Table 5.9. A motor babble and the corresponding sensory consequences. (Not all words are shown.)

	Word									
	by	*cat*	*chased*	*dog*	*meowed*	*mouse*	*the*	*thief*	*took*	*was*
Motor Babble	0.9	0.1	0.6	0.3	0.8	0.7	0.2	0.4	0.8	1.0
Sensory Consequences	0.9	0.1	0.6	0.3	0.8	0.7	0.2	0.4	0.8	1.0

The forward model is then trained using these sensory consequences as the target vector. The error for each output unit is

$$error = target - activation.$$

This function is different from the one used by the sensory network because the forward model uses linear output units. The weights are changed in the same way as in the sensory network, except that each weight is restricted to the range from -1.0 to +1.0. This keeps the weights from growing unreasonably large, as weights into linear units are prone to do.

5.4.3 Motor Network

The motor network is trained during the imitation stage, after the sensory network and forward model have already been trained. The learning situation is that a child hears a sentence that refers to the current semantic event, and he silently imitates that sentence (see figure 5.6). The input to the motor network comes from two sources. First, the child gets the semantic event from the current visual scene. This is a non-temporal signal because all of the elements of the visual scene are present at the same time. Second, the child gets the surface features for the sentence from the output of the sensory network.[1] This is a temporal signal because the sensory network activates each surface feature/value pair as its key word appears in the sentence. For example, consider the motor network input for the sentence *the cat chased the dog*, shown in table 5.10. The three active thematic role/filler pairs—Agent=*cat*, Action=*chase*, and Theme=*dog*—are all clamped on at time 1, but the sensory network does not activate the surface feature/value pair Voice=*active* until time 3, when the key word *chased* appears in the sentence.

On each time-step, the motor network is trained to produce the appropriate word of the sentence. Its target is the pattern of activations across the input units of the sensory network. This pattern represents the sensory impression formed by hearing the current word of the sentence. This target is used to train the motor network by a process called distal supervised learning (Jordan and Rumelhart 1992). First, the motor network tries to produce the current word of

[1] Actual output from the sensory network was not used during motor training. Instead, the target vector for the sensory network, which is a kind of idealized sensory output, was used. The only reason for this was expediency.

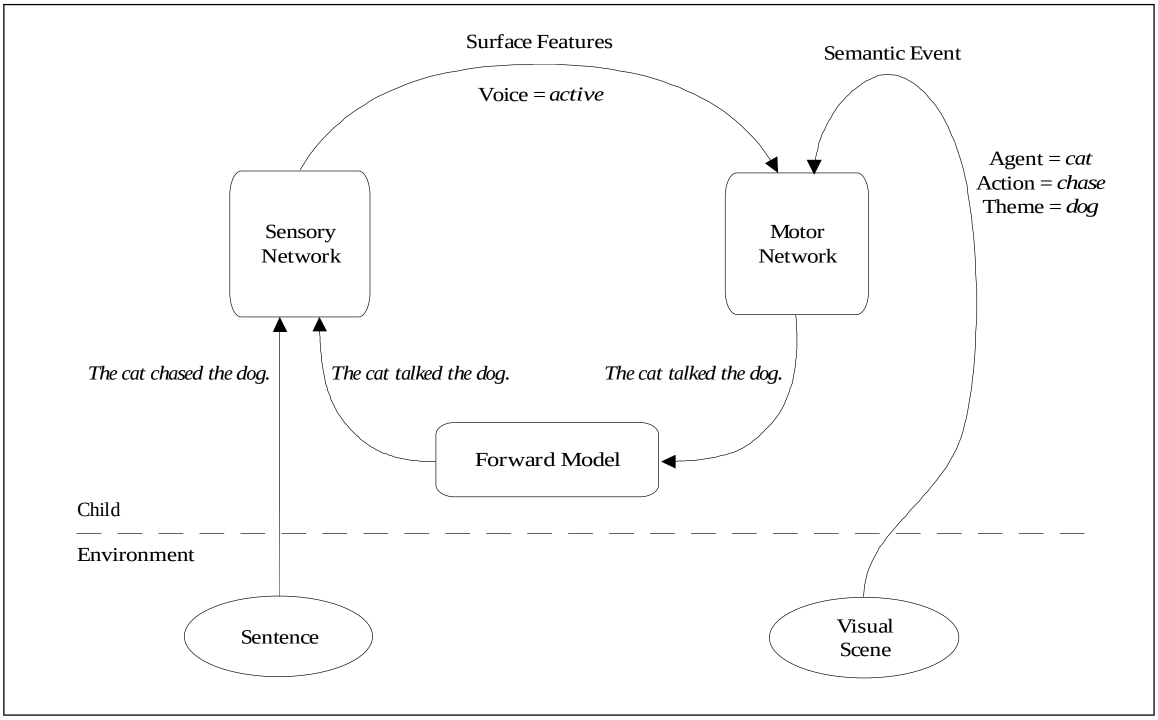

Figure 5.6. During the imitation stage, the input to the motor network comes from the current visual scene (the semantic event) and the sensory interpretation of the sentence (the surface features). The training signal is the sensory impression of the sentence, backpropagated through the forward model.

the sentence. Then, rather than being executed, the motor actions for this word are fed through the forward model. The output of the forward model is its prediction of what the word would actually sound like if it were said. This prediction is then compared to the actual sensory impression (the target) to compute the error. In the example in figure 5.6, there would be an error on time-step 3 because the motor network produced *talked* when it should have produced *chased*. Finally, this error is backpropagated through the forward model, without changing its weights, to calculate the error for each motor output

Table 5.10. The input to the motor network during training for the sentence *the cat chased the dog.* (Not all units are shown.)

	Agent			**Action**			**Theme**			**Voice**		**Prep1**	
Time	*cat*	*dog*	*thief*	*bark*	*chase*	*talk*	*cat*	*dog*	*thief*	*active*	*passive*	*by*	*with*
1	**1.0**	0.0	0.0	0.0	**1.0**	0.0	0.0	**1.0**	0.0	0.0	0.0	0.0	0.0
2	**1.0**	0.0	0.0	0.0	**1.0**	0.0	0.0	**1.0**	0.0	0.0	0.0	0.0	0.0
3	**1.0**	0.0	0.0	0.0	**1.0**	0.0	0.0	**1.0**	0.0	**1.0**	0.0	0.0	0.0
4	**1.0**	0.0	0.0	0.0	**1.0**	0.0	0.0	**1.0**	0.0	**1.0**	0.0	0.0	0.0
5	**1.0**	0.0	0.0	0.0	**1.0**	0.0	0.0	**1.0**	0.0	**1.0**	0.0	0.0	0.0

unit, and the weights in the motor network are modified using standard backpropagation.

The equations for distal supervised learning the same as those for standard backpropagation, except that the error for the motor output units is calculated using the hidden unit function

$$error = activation \cdot (1 - activation) \cdot \sum_i error_i \cdot weight_i$$

because the motor output units are acting as hidden units that feed into the forward model. The motor network uses the same learning parameters as the sensory network, a learning rate ε of 0.2 and a momentum α of 0.9.

Chapter VI. Training and Results

6.1 Training Data

The model was trained on a corpus of English sentences. These sentences were generated from semantic events in a script using a unification-based grammar. The steps in this process are depicted in figure 6.1. First, a semantic event is chosen from the script, in this case Agent=*dog*, Action=*chase*, and Theme=*cat*. Next, the event is passed though the unification-based grammar to generate all possible sentences and their surface features. In this case, there are two sentences: *the dog chased the cat* (Voice=*active*) and *the cat was chased by the dog* (Voice=*passive*, Prep1=*by*). Finally, the sentences, surface features, and semantic events are encoded as binary pattern vectors. These vectors are the training data for the computer model.

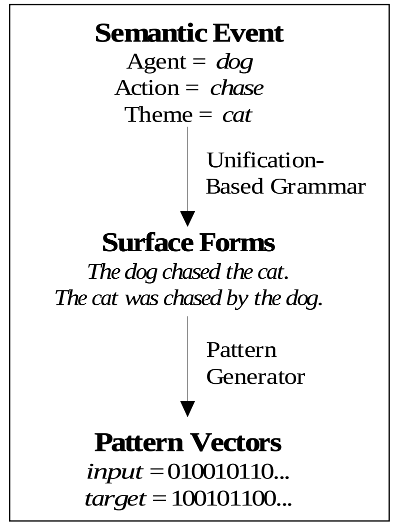

Figure 6.1. Generating a corpus of sentences and semantic events.

As shown in table 6.1, the model uses a simple set of six thematic roles. These are a subset of the thematic roles used by Parsons (1990) and Covington (1992). Every event has an Action and an Agent, and the semantics of the Action determine the other roles that are used. As indicated in the table, all of the roles other than Action and

Table 6.1. Thematic roles used in the model.

Role	*Explanation*	*Surface Location*
Action	Thing done.	Verb.
Agent	Person or thing that causes an event to occur.	Subject of an active sentence. Marked with *by* in a passive sentence.
Theme	Person or thing affected.	Direct object of an active sentence. Subject of a passive sentence.
Source	Point of origin in an action involving a transfer.	Indirect object. Marked by *from*.
Goal	Destination in an action involving a transfer.	Indirect object in an active sentence. May be subject of a passive sentence if no Theme is present. Marked by *to*.
Instrument	Means by which an action is accomplished.	Object of *with*.

Theme are marked by a unique preposition. For instance, *dog* is the Agent in the sentence *the cat was chased by the dog*. Obviously, this oversimplifies the role of prepositions in English. However, the language representation is only intended to illustrate how the computer model works; it is not meant to be an accurate model of English in its own right.

The unification-based grammar that translates semantic events into sentences is encoded in a Prolog and GULP (Covington 1992) program. This program generates all possible sentences for each event. The sentences are third-person, past-tense, and singular. Each contains one or more noun phrases, a verb, and one or more prepositional phrases. The general form of the grammar, without feature structures, is as follows:

$$S \rightarrow NP\ VP$$
$$NP \rightarrow D\ N$$
$$VP \rightarrow V$$
$$VP \rightarrow V\ NP$$
$$VP \rightarrow V\ NP\ NP$$
$$VP \rightarrow V\ NP\ PP$$
$$VP \rightarrow V\ PP\ PP$$
$$VP \rightarrow V\ NP\ PP\ PP$$

Two sources of variation make it possible for each event to have several equivalent surface forms:

1. *Active and passive voice*. Every event has both an active and a passive voice surface form. For example, the event Action=*talk*, Agent=*waitress*, and Goal=*cat* can be translated into either *the waitress talked to the cat* or *the cat was talked to by the waitress.*
2. *Variable prepositional phrase order*. The order of prepositional phrases is arbitrary. For instance, the event Action=*hit*, Agent=*waitress*, Theme=*thief*, and Instrument=*purse* can be translated into both *the thief was hit by the waitress with the purse* and *the thief was hit with the purse by the waitress.*

Some events have as many as five equivalent surface forms.

6.2 Training Methods and Results

The unification-based grammar generated 192 sentences from a script containing 86 events. Of the 192 sentences, 20 were selected randomly and set aside in a test corpus that was not used for training. The remaining 172 sentences constituted the training corpus. The model was then trained in stages, as described in section 5.4.

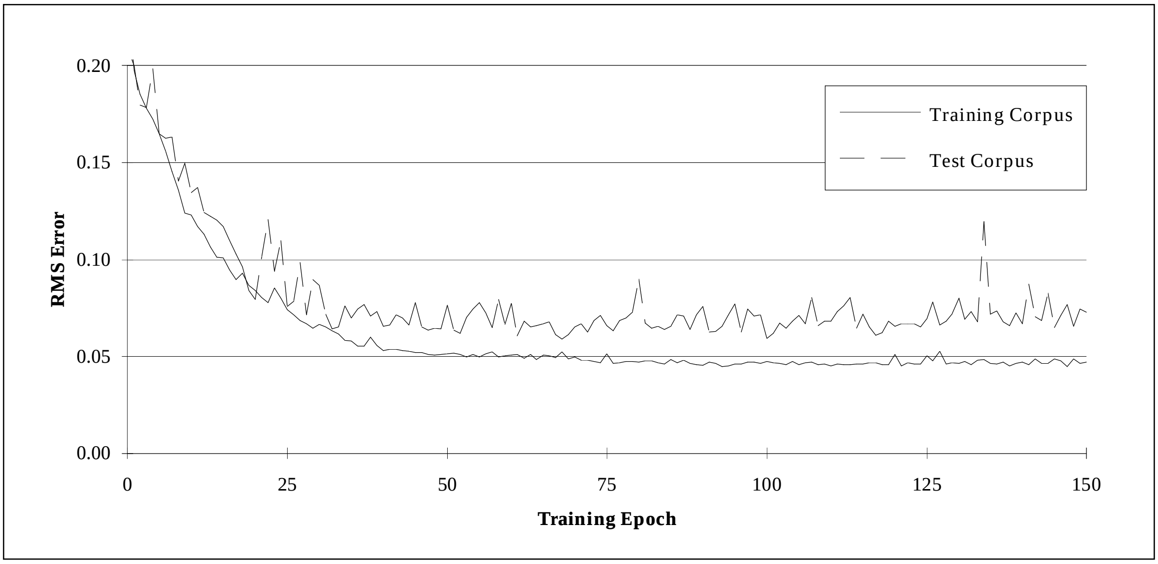

Figure 6.2. The learning performance of the sensory network as measured by RMS error.

6.2.1 Sensory Network

The sensory network was trained for 150 epochs. Longer training runs were tried, but they did not improve its performance. Each epoch consisted of a single pass through the entire training corpus in a random order. Duplicate presentations were allowed, so some sentences might be presented twice on one epoch and others not at all.

On each epoch, the network performance was evaluated in two ways. First, the root-mean-squared (RMS) error was recorded for both the training and test corpuses (see figure 6.2). This statistic measures the distance between the target and output vectors for a pattern. The formula is

$$RMS\ error = \sqrt{\frac{1}{n}\sum_i (target_i - output_i)^2}$$

where *i* ranges over all *n* of the output units. The RMS error for an entire epoch is the average error for all of the patterns in the corpus.

RMS error is somewhat hard to interpret, so the sensory network was also tested to see how many output errors it made on each corpus. An output error was defined to be an output unit activation of either less than 0.7 for a unit that should be on or greater than 0.3 for a unit that should be off. This corresponds to a mistake in instantiating either a thematic role/filler unit or a surface feature/value unit. Errors were measured after the last time-step in each sentence, and the total number of errors was counted for each training epoch. Then the total number of errors for each corpus was normalized by dividing by the total number of instantiations in all of the sentences in the corpus. For example, the sentence *the waitress kissed the cat* has four instantiations: Agent=*waitress*, Action=*kiss*, Theme=*cat*, and Voice=*active*. So it would contribute

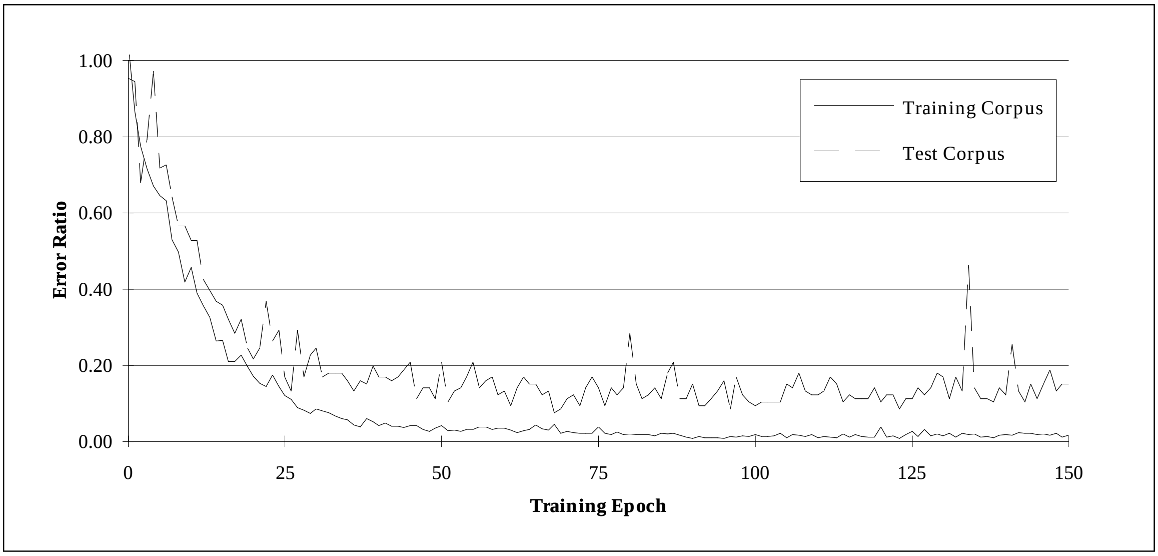

Figure 6.3. The learning performance of the sensory network as measured by the error ratio for each corpus.

four instantiations to the total for its corpus. The normalized score is referred to as the error ratio (see figure 6.3).

6.2.2 Forward Model

The forward model was trained until it produced an RMS error of less than 1.0×10^{-6}, which is essentially perfect performance. This took 670 epochs, where an epoch is a single presentation of a one-word babble.

6.2.3 Motor Network

The motor network was trained in the same way as the sensory network. Its learning performance as measured by its RMS error is shown in figure 6.4. Output errors were also recorded for each corpus using the same cutoffs as in the sensory network. In the motor network, an output error corresponds to a mistake in producing a word. For example, if the motor network produced the sentence *the thief talked to the cat* when the target was *the thief talked to the waitress*, that would be one output error. Output errors were counted after every time-step in each sentence. Then the total number of errors for each corpus was normalized by dividing by the total number of words in all of the sentences in the corpus. The resulting error ratios are shown in figure 6.5.

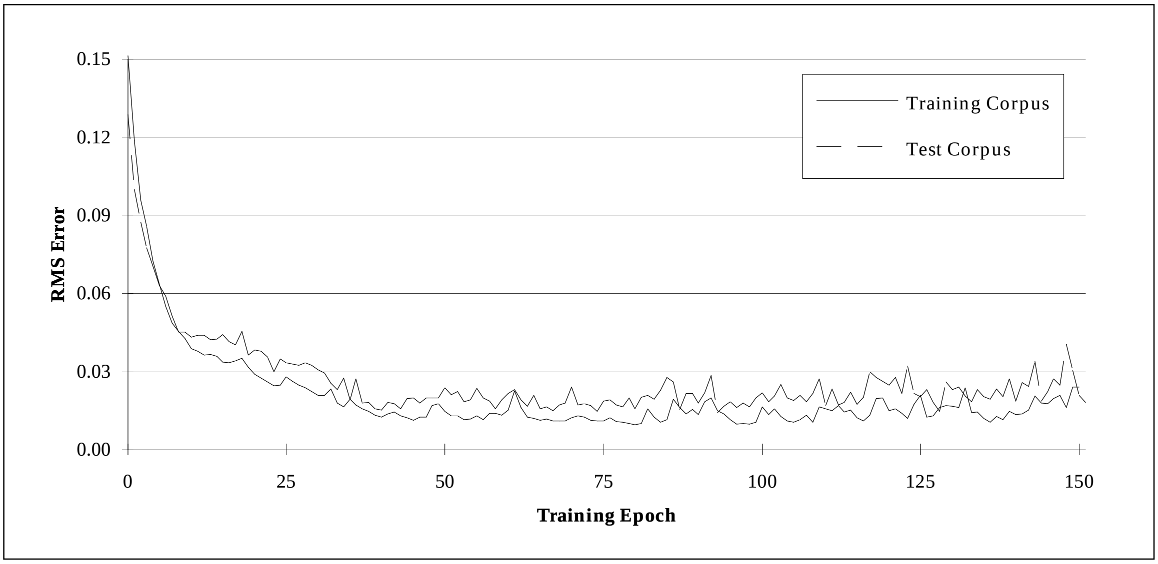

Figure 6.4. The learning performance of the motor network as measured by RMS error.

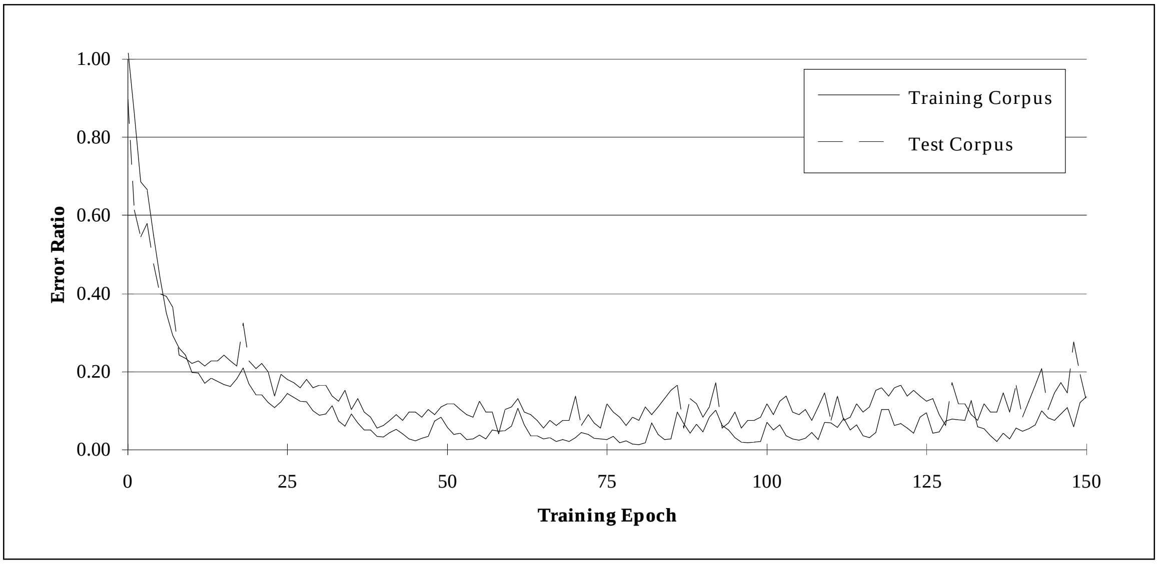

Figure 6.5. The learning performance of the motor network as measured by the error ratio for each corpus.

Chapter VII. Conclusion

7.1 Evaluation

In evaluating the model, it is important to keep in mind its goals. The sensorimotor preadaptation hypothesis predicts that the prior existence of advanced cognition preadapted general sensory and motor mechanisms for language. The model's purpose is to support this hypothesis by showing how general sensory and motor mechanisms might produce language. I have suggested what these general sensory and motor mechanisms are, and explained how they are simulated in the model. The remaining question is, Does the model's performance justify the claim that it has produced language?

The two main parts of the model, the sensory and motor networks, both learn to perform well. At the point where the sensory network reached its lowest RMS error (epoch 93), it had an error ratio of 0.010 for the training corpus and 0.113 for the test corpus. This means that it instantiated 99.0% of the thematic role / filler and surface feature / value pairs correctly for sentences in the training corpus and 88.7% correctly for sentences in the test corpus. Likewise, at the point where the motor network reached its lowest RMS error (epoch 80), it had an error ratio of 0.012 for the training corpus and 0.076 for the test corpus. This means that it instantiated 98.8% of the words correctly for sentences in the training corpus and 92.4% correctly for sentences in the test corpus.

The model's performance on the training corpus is clearly very good. Both the sensory and motor networks produced the correct instantiation about 99% of the time. People probably misunderstand a word or speak incorrectly at least 1% of the time, so this is an adequate level of performance. On the other hand, performance on the test corpus was not as good. The sensory network produced an incorrect instantiation about 11% of the time, and the motor network about 8%. Obviously, people do much better than this on novel sentences, so the model does not fully account for language generalization.

However, it is not really a surprise that the model does not generalize as well as people do. The generalization performance of connectionist models depends heavily on the network architecture used. The sensory and motor networks use a very simple architecture that is known to have limited generalization abilities (Mozer 1993). The human neocortex is much more complex. Presumably, people generalize so well because the architecture that emerges from all of this complexity has very powerful generalization abilities. If so, then reproducing human-level generalization is really beyond the scope of this model. The purpose of the model is to show how general sensory and motor mechanisms might produce language, not to mimic the local circuity of the neocortex. The model's generalization ability can probably be improved by adopting a more powerful network architecture like de Vries and Principe's (1992) gamma model, but such changes really don't really improve the model as a simulation of sensory and motor processing in the neocortex. Instead, it seems more reasonable to conclude that the model shows that it is possible that language is produced by general sensory and motor mechanisms, but a full account of some aspects of language, like generalization, will have to wait until know more about the details of neocortical processing.

7.2 Future Directions

One area where the model could be improved is in its simulation of sensory and motor mechanisms. Currently, the sensory and motor components of the model each use a single recurrent network to simulate the suggested role of two brain networks, a topographic map and an associative network. It would be nice to eliminate this extra layer of abstraction by incorporating the hypothesized brain networks directly into the model. The primary obstacle is that there are no connectionist techniques for forming temporal or motor topographic maps. Perhaps a reasonable goal for future research would be to devise a network architecture capable of forming either sort of topographic map and incorporate it into the model.

Another area for future research is linking the model's sensory and motor networks to particular brain areas. Cognitive neuroscience techniques like PET scans, functional MRI, EEGs, and lesion analyses have all been used to localize various aspects of language processing in the brain. I have not incorporated this evidence into the model because I did not wish to complicate it further. However, in the long run, it is essential to link hypothesized neural systems to particular brain areas. Such links make more detailed neural constraints available to the modeler, and they provide a way to test the model, by lesioning particular model components and comparing the results to equivalent lesions in clinical subjects.

Bibliography

Allman, John (1990). Evolution of neocortex. In *Cerebral cortex*, Vol. 8A, *Comparative structure and evolution of cerebral cortex: Part I*, eds. E. G. Jones and A. Peters. New York: Plenum.

Bellugi, Ursula, Howard Poizner, and Edward S. Klima (1989). Language, modality and the brain. *Trends in Neurosciences* **12**:380-388.

Bellugi, Ursula, Howard Poizner, and Edward S. Klima (1989). Language, modality and the brain. *Trends in Neurosciences* **12**:380-388.

Britton, Bruce K., and F. James Eisenhart (in press). Expertise, text coherence and constraint satisfaction: Effects on harmony and settling rate. In *Proceedings of the fifteenth annual meeting of the Cognitive Science Society*. Boulder, CO.

Chomsky, Noam (1980). Rules and representations. *Behavioral and Brain Sciences* **3**:1-61.

Cottrell, Garrison W., and Steven L. Small (1983). A connectionist scheme for modeling word sense disambiguation. *Cognition and Brain Theory* **6**:89-120.

Covington, Michael A. (1992). *Natural language processing for Prolog programmers.* Publication draft. To be published by Prentice-Hall, 1993.

de Vries, Bert, and Jose C. Principe (1992). The gamma model—A new neural net model for temporal processing. *Neural Networks* **5**:565-576.

Donald, Merlin (1991). *Origins of the modern mind: Three stages in the evolution of culture and cognition.* Cambridge: Harvard University Press.

Elman, Jeffrey L. (1990). Finding structure in time. *Cognitive Science* **14**:179-211.

Fillmore, Charles J. (1968). The case for case. In *Universals in linguistic theory*, eds. E. Bach and R. T. Harms, 1-88. New York: Rinehart and Winston.

Hebb, Donald H. (1949). *The organization of behavior: A neuropsychological theory.* New York: Wiley.

Hinton, Geoffrey. E., and Terrence J. Sejnowski (1986). Learning and relearning in Boltzmann machines. In Rumelhart, McClelland, and the PDP Research Group (1986a), 282-317.

Jain, Ajay N. (1991). PARSEC: A connectionist learning architecture for parsing spoken language. Ph.D. thesis CMU-CS-91-208, School of Computer Science, Carnegie Mellon University. Pittsburgh, PA.

Jordan, Michael I. (1986). Attractor dynamics and parallelism in a connectionist sequential machine. In *Proceedings of the eighth annual conference of the Cognitive Science Society*, 531-546. Amherst, MA.

Jordan, Michael I., and David E. Rumelhart (1992). Forward models: Supervised learning with a distal teacher. *Cognitive Science* **16**:307-354.

Kaas, Jon H. (1987). The organization of neocortex in mammals: Implications for theories of brain function. *Annual Review of Neuroscience* **38**:129-151.

Kandel, Eric R., James H. Schwartz, and Thomas M. Jessell (1991). *Principles of neural science*. 3d ed. New York: Elsevier.

Kimura, D. (1979). Neuromotor mechanisms in the evolution of human communication. In *Neurobiology of social communication in primates*, eds. H. D. Steklis and M. J. Raleigh. New York: Academic Press.

Kintsch, Walter (1988). The role of knowledge in discourse comprehension: A construction-integration model. *Psychological Review* **95**:163-182.

Kintsch, Walter, and David M. Welsch (1990). The construction-integration model: A framework for studying memory for text. Institute for Cognitive Science Research Report 90-15, University of Colorado. Boulder, CO.

Konishi, Masakazu (1985). Birdsong: From behavior to neuron. *Annual Review of Neuroscience* **8**:125-170.

Lieberman, Philip (1984). *The biology and evolution of language*. Cambridge: Harvard University Press.

Linsker, Ralph (1986). From basic network principles to neural architecture. Series of 3 articles. *Proceedings of the National Academy of Science* **83**:7508-7512, 8390-8394, 8779-8783.

Margoliash, Daniel (1983). Acoustic parameters underlying the responses of song-specific neurons in the white-crowned sparrow. *The Journal of Neuroscience* **3**:1039-1057.

McClelland, James L., and Alan H. Kawamoto (1986). Mechanisms of sentence processing: Assigning roles to constituents of sentences. In Rumelhart, McClelland, and the PDP Research Group (1986b), 272-325.

McClelland, James L., and David E. Rumelhart (1981). An interactive activation model of context effects in letter perception: Part I. An account of basic findings. *Psychological Review* **88**:375-407.

Mozer, Michael C. (1993). Neural net architectures for temporal sequence processing. Preprint. To appear in *Predicting the future and understanding the past*, eds. A. Weigend and N. Gershenfeld. Redwood City, CA: Addison-Wesley.

Oja, Erkki (1982). A simplified neuron model as a principal component analyzer. *Journal of Mathematical Biology* **15**:267-273.

Parsons, Terrence (1990). *Events in the semantics of English: A study in subatomic semantics*. Cambridge: MIT Press.

Rosenblatt, Frank (1962). *Principles of neurodynamics*. New York: Spartan.

Rumelhart, David E., Geoffrey E. Hinton, and Ronald J. Williams (1986). Learning internal representations by error propagation. In Rumelhart, McClelland, and the PDP Research Group (1986a), 318-364.

Rumelhart, David E., and James L. McClelland (1986). On learning the past tenses of English verbs. In Rumelhart, McClelland, and the PDP Research Group (1986b), 216-271.

Rumelhart, David E., James L. McClelland, and the PDP Research Group (1986a). *Parallel distributed processing: Explorations in the microstructure of cognition, Vol. 1: Foundations.* Cambridge: MIT Press.

Rumelhart, David E., James L. McClelland, and the PDP Research Group (1986b). *Parallel distributed processing: Explorations in the microstructure of cognition, Vol. 2: Psychological and biological models.* Cambridge: MIT Press.

Rumelhart, David E., and David Zipser (1985). Feature discovery by competitive learning. *Cognitive Science* **9**:75-112. Reprinted in Rumelhart, McClelland, and the PDP Research Group (1986a), 151-193.

Sarich, V. M. (1980). Molecular clocks and hominid evolution after 12 years. *American Journal of Physical Anthropology* **52**:275-276.

Shepherd, Gordon M. (1988). A basic circuit of cortical organization. In *Perspectives in memory research*, ed. M. S. Gazzaniga, 93-134. Cambridge, MA: Bradford.

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

St. John, Mark F. (1992a). The story gestalt: A model of knowledge-intensive processes in text comprehension. *Cognitive Science* **16**:271-306.

St. John, Mark F. (1992b). Learning language in the service of a task. In *Proceedings of the fourteenth annual conference of the Cognitive Science Society*, 271-276. Hillsdale, NJ: Erlbaum.

Suga, Nagumo. (1988). Auditory neuroethology and speech processing: Complex-sound processing by combination sensitive neurons. In *Auditory function: Neurobiological bases of hearing*, eds. G. M. Edelman, W. E. Gall, and R. W. M. Cowan, 679-720. New York: Wiley.

von der Malsburg, Christoph (1973). Self-organization of orientation sensitive cells in the striate cortex. *Kybernetik* **14**:85-100. Reprinted in *Pattern recognition by self-organizing neural networks*, eds. G. A. Carpenter and S. Grossberg, 180-205. Cambridge, MA: Bradford, 1991.

Waltz, David L., and Jordan B. Pollack (1985). Massively parallel parsing: A strongly interactive model of natural language interpretation. *Cognitive Science* **9**:51-74.

Wickelgren, W. A. (1969). Context-sensitive coding, associative memory, and serial order in (speech) behavior. *Psychological Review* **86**:44-60.

Williams, Ronald J., and David Zipser (1989). A learning algorithm for continually running fully recurrent neural networks. *Neural Computation* **1**:270-280.