LEARNING AND APPLYING CONTEXTUAL CONSTRAINTS IN SENTENCE COMPREHENSION

Technical Report AIP - 39

M. F. St. John & J. L. McClelland

Carnegie Mellon University
Department of Psychology
Pittsburgh, Pa., 15213

8 June 1988

The authors would like to thank Geoffrey Hinton, Brian MacWhinney, Andrew Hudson, and the members of the PDP Research Group at Carnegie Mellon. This research was supported by the Computer Sciences Division, Office of Naval Research and DARPA under Contract Number N00014-86-K-0678, ONR Contracts N00014-86-G-0146, N00014-86-K-00167 and N00014-86-K-0349; NSF grant BNS 86-09729 and NIMH Career Development Award MH00385 to the second author. Reproduction in whole or in part is permitted for purposes of the United States Government. Approved for public release; distribution unlimited.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; Distribution unlimited
2b. DECLASSIFICATION / DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) AIP - 39	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Carnegie-Mellon University 6b. OFFICE SYMBOL (*If applicable*)	7a. NAME OF MONITORING ORGANIZATION Computer Sciences Division Office of Naval Research
6c. ADDRESS (*City, State, and ZIP Code*) Department of Psychology Pittsburgh, Pennsylvania 15213	7b. ADDRESS (*City, State, and ZIP Code*) 800 N. Quincy Street Arlington, Virginia 22217-5000
8a. NAME OF FUNDING / SPONSORING ORGANIZATION Same as Monitoring Organization 8b. OFFICE SYMBOL (*If applicable*)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER N00014-86-K-0678
8c. ADDRESS (*City, State, and ZIP Code*)	10. SOURCE OF FUNDING NUMBERS p4000ub201/7-4-86 PROGRAM ELEMENT NO: N/A; PROJECT NO.: N/A; TASK NO.: N/A; WORK UNIT ACCESSION NO: N/A
11. TITLE (*Include Security Classification*) Learning and applying contextual constraints in sentence comprehension	
12. PERSONAL AUTHOR(S) Mark F. St. John and James L. McClelland	
13a. TYPE OF REPORT Technical	13b. TIME COVERED FROM 86Sept15 TO 91Sept14
14. DATE OF REPORT (*Year, Month, Day*) 1988 June 8	15. PAGE COUNT 47
16. SUPPLEMENTARY NOTATION	

17. COSATI CODES		
FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS (*Continue on reverse if necessary and identify by block number*)
Language understanding, Connectionist models, Ambiguity resolution

19. ABSTRACT (*Continue on reverse if necessary and identify by block number*)

A parallel distributed processing model is described that learns to comprehend single clause sentences. Specifically, it assigns thematic roles to sentence constituents, disambiguates ambiguous words, instantiates vague words, and elaborates implied roles. The sentences are pre-segmented into constituent phrases. Each constituent is processed in turn to update an evolving representation of the event described by the sentence. The model uses the information derived from each constituent to revise its on-going interpretation of the sentence and to anticipate additional constituents. The network learns to perform these tasks through practice on processing example sentence/event pairs. The learning procedure allows the model to take a long-range statistical approach to solving the bootstrapping problem of learning the syntax and semantics of a language from the same data. The model performs very well on the corpus of sentences on which it was trained, but learns slowly.

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT ☐ UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT. ☐ DTIC USERS	21 ABSTRACT SECURITY CLASSIFICATION	
22a NAME OF RESPONSIBLE INDIVIDUAL Dr. Alan L. Meyrowitz	22b TELEPHONE (*Include Area Code*) (202) 696-4302	22c. OFFICE SYMBOL N00014

DD FORM 1473, 84 MAR 83 APR edition may be used until exhausted. All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE

Unclassified

Abstract

A parallel distributed processing model is described that learns to comprehend single clause sentences. Specifically, it assigns thematic roles to sentence constituents, disambiguates ambiguous words, instantiates vague words, and elaborates implied roles. The sentences are pre-segmented into constituent phrases. Each constituent is processed in turn to update an evolving representation of the event described by the sentence. The model uses the information derived from each constituent to revise its on-going interpretation of the sentence and to anticipate additional constituents. The network learns to perform these tasks through practice on processing example sentence/event pairs. The learning procedure allows the model to take a long-range statistical approach to solving the bootstrapping problem of learning the syntax and semantics of a language from the same data. The model performs very well on the corpus of sentences on which it was trained, but learns slowly.

The goal of our research has been to develop a model that can learn to convert a simple sentence into a conceptual representation of the event that the sentence describes. Specifically, we have been concerned with the later stages of this process: the conversion of a sequence of sentence constituents, such as noun phrases, into a representation of the event. A number of problems make this process difficult. First, the words of a sentence may be ambiguous or vague. In the sentence, "The pitcher threw the ball," each content word is ambiguous. "Pitcher" could either refer to a ball-player or a container; "threw" could either refer to toss or host; and "ball" could refer to a sphere or a dance. How are the appropriate meanings selected so that a single, coherent interpretation of the sentence is produced? Vague words also present difficulties. In the sentences, "The container held the apples" and "The container held the cola," the word "container" refers to two different objects (Anderson & Ortony, 1975). How does the context affect the interpretation of vague words?

A third problem is the complexity of assigning the correct thematic roles (Fillmore, 1968) to the objects referred to in a sentence. Consider

1) The teacher ate the spaghetti with the busdriver.

2) The teacher ate the spaghetti with the red sauce.

3) The busdriver hit the fireman.

4) The busdriver was hit by the fireman.

In the first two examples, semantics play an important role. In the first sentence, it is the reader's knowledge that busdrivers are people that precludes the reader from deciding the busdriver is to be served as a condiment. Instead, it must be that both he and the teacher are eating the spaghetti. Semantic constraints work conversely in the

second sentence. In the third sentence, semantics do not help determine who is the agent and who is the patient. Instead, word order determines the thematic role assignments. The busdriver is the agent because "the busdriver" is the pre-verbal constituent. Finally, in the fourth sentence, the influence of other morphological features can be seen. The passive verb tense and the "by" preposition, in conjunction with the word order, determine that the busdriver is the patient. Thematic role assignment, then, requires the joint consideration of a variety of aspects of the sentence.

A fourth problem for processing sentences is that a sentence may leave some thematic constituents implicit that are nevertheless present in the event. For example in sentences 1 and 2 above, the spaghetti was undoubtedly eaten with forks. Psychological evidence indicates that missing constituents, when strongly related to the action, are inferred and added to the description of the event. McKoon and Ratcliff (1981) found, for example, that "hammer" was inferred after subjects read "Bobby pounded the boards together with nails."

Our model of the comprehension process centers on viewing the process as a form of constraint satisfaction. The surface features of a sentence, its particular words and their order and morphology, provide a rich set of constraints on the sentence's meaning. Each feature constrains the meaning in a number of respects. Conjunctions of features, such as word order and passive-voice morphology, provide additional constraints. Together, the constraints lead to a coherent interpretation of the sentence (MacWhinney, 1987). These constraints are not typically all-or-none. Instead, constraints tend to vary in strength: some are strong and others are relatively weak. An example adapted from Marcus (1980) provides a good illustration of the competition between constraints.

1) Which dragon did the knight give the boy?

2) Which boy did the knight give the dragon?

3) Which boy did the knight give the sword?

4) Which boy did the knight give to the sword?

Apparently, in the first two sentences, a weak syntactic constraint makes us prefer the first noun as the patient and the noun after the verb as the recipient. The subtle semantics in the second sentence, that knights don't give boys to dragons, does not override the syntactic constraint for most readers, though it may make the sentence seem ungrammatical to some. In sentence 3, a stronger semantic constraint overrides this syntactic constraint: swords, which are inanimate objects, cannot receive boys. Finally, in the fourth sentence, a stronger syntactic constraint overrides the semantics. It is clear from this example that constraints vary in strength and compete to produce an interpretation of a sentence. A good method for capturing this competition is to assign real-valued strengths to the constraints, and to allow them to compete or cooperate according to their strength.

Parallel distributed processing, or connectionist, models are particularly good for modeling this style of processing. They allow large amounts of information to be processed simultaneously and competitively, and they allow evidence to be weighted on a continuum (McClelland & Rumelhart, 1981; McClelland & Elman, 1986). A number of researchers have pursued this idea and have built models to apply connectionism to sentence processing (Cottrell, 1985; Cottrell & Small, 1983; Waltz & Pollack, 1985). The development of this approach, however, has been retarded because it is difficult to determine exactly what constraints are imposed by each feature or set of features in a

sentence. It is even more difficult to determine the appropriate strengths each of these constraints should have. Connectionist learning procedures, however, allow a model to learn the appropriate constraints and assign appropriate strengths to them.

To take advantage of this feature, learning was added to our list of goals. The model is given a sentence as input. From the sentence, the model must produce a representation of the event to which the sentence refers. The actual event that corresponds to the sentence is then used as feedback to train the model. But learning is not without its own problems. Several features of the learning task make learning difficult. One problem is that the environment is probabilistic. On different occasions, a sentence may refer to different events: it may be referentially ambiguous. For example, a sentence like, "The pitcher threw the ball," may refer to either the tossing of a projectile or the hosting of a party. The robust, graded, and incremental character of connectionist learning algorithms leads us to hope that they will be able to cope with the variability in the environment in which they learn.

A second learning problem concerns the difficulty of learning the mapping between the parts of the sentence and the parts of the event (Gleitman & Wanner, 1982; Quine, 1960). Learning the mapping is sometimes referred to as a boot-strapping problem since the meaning of the content words and significance of the syntax must be acquired from the same set of data. To learn the syntax, it seems necessary to already know the word meanings. Conversely, to learn the word meanings it seems necessary to know how the syntax maps the words onto the event description. The connectionist learning procedure takes a statistical approach to this problem. Through exposure to large numbers of sentences and the events they describe, the mapping between features of the sentences and characteristics of the events will emerge as statistical regularities.

For instance, in the long run the learning procedure should discover the regularity that sentences beginning with "the boy" and containing a transitive verb in the active voice refer to events in which a young, male human participates as an agent. The discovery of the entire ensemble of such regularities provides a joint solution to the problems of learning the mapping and the meanings of words.

Some aspects of these goals have been addressed by our own earlier work (McClelland & Kawamoto, 1986; St. John & McClelland, 1987). However, these previous models used a cumbersome *a priori* representation of sentences that proved unworkable (see St. John & McClelland, 1987 for discussion). Given the recent successes in using connectionist learning procedures to learn internal representations (Hinton, 1986; Rumelhart, Hinton, & Williams, 1986), we decided to explore the feasibility of having a network learn its own representation of sentences.

A final characteristic of language comprehension we wanted to capture is sometimes called the principle of immediate update (Carpenter & Just, 1977; van Dijk & Kintsch, 1983; Marslen-Wilson & Tyler, 1980). As each constituent of the sentence is encountered, the interpretation of the entire event is adjusted to reflect the constraints arising from the new constituent in conjunction with the constraints from constituents already encountered. Based on all of the available constraints, the model should try to anticipate upcoming constituents. It should also adjust its interpretation of preceding constituents to reflect each new bit of information. In this way, particular sentence interpretations may gain and lose support throughout the course of processing as each new bit of information is processed. This immediate update should be accomplished while avoiding the difficulty of performing backtracking.

In sum, the model addresses six goals:

* to disambiguate ambiguous words

* to instantiate vague words

* to assign thematic roles

* to elaborate implied roles

* to learn to perform these tasks

* to immediately adjust its interpretation

as each constituent is processed

Description of the SG Model

Task

The model's task is to process a single clause sentence without embeddings into a representation of the event it describes. The sentence is presented to the model as a temporal sequence of constituents. A constituent is either a simple noun phrase, a prepositional phrase, or a verb (including the auxiliary verb, if any). The information each of these sentence constituents yields is immediately used to update the model's internal representation of the event. This representation is called the sentence gestalt because all of the information from the sentence is represented together within a single, distributed representation; the model is called the Sentence Gestalt, SG, model because it contains this representation. This general concept of sentence representation comes from Hinton's pioneering work (Hinton, 1981). From the sentence gestalt, the model can produce, as output, a representation of the event. This event representation consists of a set of pairs. Each pair consists of a thematic role and the concept that fills that role. Together, the pairs describe the event.

Architecture and processing

The model consists of two parts. One part, the sequential encoder, sequentially processes each constituent to produce the sentence gestalt. The second part is used to produce the output representation from the sentence gestalt.

Producing the sentence gestalt. To process the constituent phrases of a sentence, we adapted an architecture from Jordan (1986) that uses the output of previous processing as input on the next iteration (see Figure 1). Each constituent is processed in turn to update the sentence gestalt. To process a constituent, it is first represented as a pattern of activation over the *current constituent* units. Activation from these units projects to a hidden unit layer and combines with the activation from the *sentence gestalt* units created as the result of processing the previous constituent. The actual implementation of this arrangement is to copy the activation from the *sentence gestalt* to the *previous sentence gestalt* units, and allow activation to feed

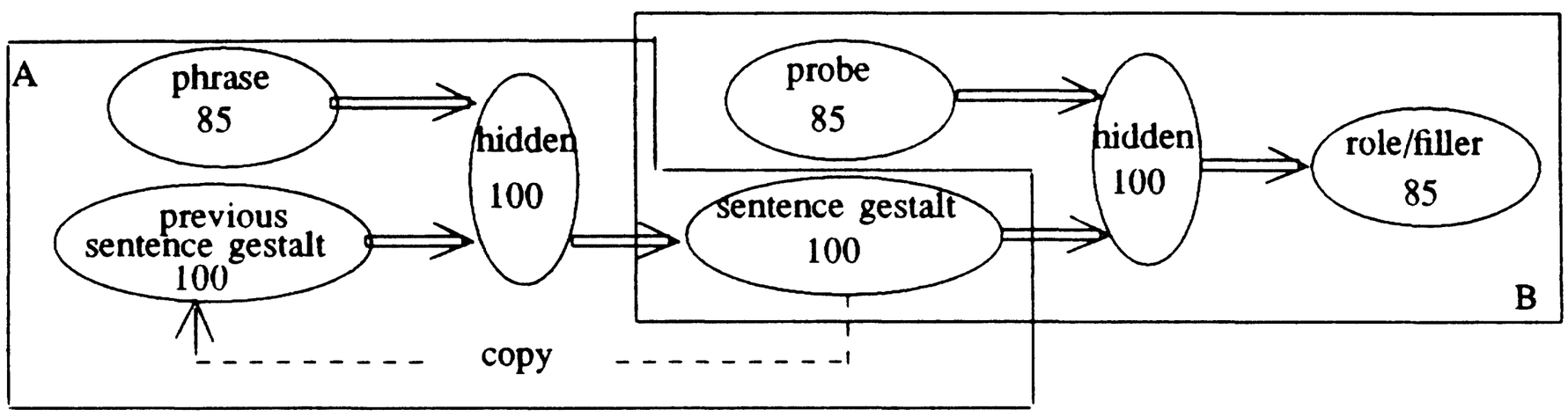


Figure 1. The architecture of the network. The boxes highlight the functional parts: Area A processes the phrases into the sentence gestalt, and Area B processes the sentence gestalt into the output representation. The numbers indicate the number of units in each layer.

forward from there. Activation in the hidden layer then creates a new pattern of activation over the *sentence gestalt* units.

Producing the output. A noted previously, several other models have used a type of sentence gestalt to represent a sentence. McClelland and Kawamoto (1986) used units that represented the conjunction of semantic features of the verb with the semantic features of a concept. To encode a sentence, the patterns of activity produced for each verb/concept were activated in a single pool of units that contained every possible conjunction. St. John and McClelland (1987) used a similar conjunctive representation to encode a number of sentences at once. These representations suffer from inefficiency and scale badly because so many units are required to represent all of the conjunctions. The current model's representation is far more efficient.

The model's efficiency comes from making the sentence gestalt a trainable, hidden unit layer. Making the sentence gestalt trainable allows the network to create the primitives it needs to represent the sentence efficiently. Instead of having to represent every possible conjunction, only those conjunctions that are useful will be learned and added to the representation. Further, these primitives do not have to be conjunctions between the verb and a concept. A hidden layer could learn to represent conjunctions between the concepts themselves or other combinations of information if they were useful for solving its task.

Since a layer of hidden units cannot be trained directly, we invented a way of "decoding" the sentence gestalt into an output layer. The output layer represents the event as a set of thematic role and filler pairs. For example, the event described by "The pitcher threw the ball" would be represented as the set {agent/pitcher(ball-player), action/threw(toss), patient/ball(sphere)}.

The output layer can represent one role/filler pair at a time. To decode a particular role/filler pair, the sentence gestalt is probed with half of the pair. Activation from the probe and the *sentence gestalt* combine in another hidden layer which then activates the entire pair in the output layer. The entire event can be decoded in this way by successively probing with each half of each pair.

When more than one concept can plausibly fill a role, we assume that the correct response is to activate each possible filler to a degree. The degree of activation of the units representing each filler corresponds to the filler's conditional probability of occurring in the given context. The network should learn weights to produce these activations through training. To achieve this goal, we employed an error measure in the learning procedure, cross-entropy (Hinton, 1987), that converges on this goal:

$$C = -\sum_j [T_j \log_2 (A_j) + (1-T_j) \log_2 (1-A_j)]$$

where T_j is the target activation and A_j is the output activation of unit j. As with many connectionist learning procedures, the goal is to minimize the error measure or cost-function (cf. Hinton, 1987). The minimum of C occurs at the point in weight space where the activation value of each output unit equals the conditional probability that the unit should be on in the current context. In the model, when the network is probed with a particular role, several of the output units represent the occurrence of a particular filler of that role. When C is at its minimum, the units' activation values represent the conditional probability of the occurrence of that filler, in that role, given the current situation.[1] Probing with the filler works similarly. The activation value of each role unit in the output layer represents the conditional probability of the probed filler playing that role in the current situation. In performing gradient descent in C, the network is

searching for weights that allow it to match activations to these conditional probabilities.

Environment and training regime

Training consists of trials in which the network is presented with a sentence and the event it describes. These sentence/event pairs were generated on-line for each training trial. Some pairs were more likely to be generated than others. Over training, these likelihood differences translated into differences in training frequency.

The network is trained to generate the event from the sentence as input. To promote immediate processing, a special training regime is used. After each constituent has been processed, the network is trained to predict the set of role/filler pairs of the entire sentence. From the first constituent of the sentence, then, the model is forced to try to predict the entire event. This training regime, therefore, assumes that the complete event is available to the learning procedure as soon as sentence processing begins, but it does not assume any special knowledge about which aspects of the event correspond to which sentence constituents. Of course, after processing only the first constituent, the model generally cannot correctly guess the entire event. By forcing it to try, this training procedure requires the model to discover the mapping between constituents and aspects of the event, as it forces the model to extract as much information as possible from each constituent. Consequently, as each new constituent is processed, the model's predictions of the event are refined to reflect the additional evidence it supplies.

An illustration of processing

An example of how a trained network processes a sentence will help illustrate

how it works. To process the sentence, "The teacher ate the soup," the constituents of the sentence are processed in turn. As each constituent is processed, the networks performs a type of pattern completion. The model augments the information supplied by each constituent with additional information about the event.

With each additional constituent, the model's predictions improve. Early in the sentence, many possible events are consistent with what little is known about the sentence so far. The completion process activates each of these alternatives slightly, according to their support. As more constituents are processed, the additional evidence more strongly supports fewer possible events.

The pattern of activation over the sentence gestalt can be observed directly, and responses to probes can be examined, to see what it is representing after processing each constituent of the sentence (see Figure 2). After processing the first constituent, "The teacher" of our example sentence, the network assumes the sentence is in the active voice and therefore assigns *teacher* to the agent role. The network also fills in the semantic features of teachers according to its previous experience (i.e. person, adult, and female). When probed with the action role, the network weakly activates a number of possible actions which the teacher performs. The network similarly makes guesses about the other roles for which it is probed.

When the second constituent, "ate," is processed, the sentence gestalt is refined to represent the new information. In addition to representing both that *teacher* is the agent and that *ate* is the action, the network is able to make better guesses about the other roles. For example, it infers that the patient is food. Since, in the network's experience, teachers typically eat soup, the network produces activation corresponding to the inference that the food is *soup*. After the third constituent is processed, the

---

Figure 2 - Sentence Gestalt evolution

see following page

Figure 2. The evolution of the sentence gestalt during processing. On the left, the activation of part of the sentence gestalt is shown after each sentence constituent has been processed. On the right, the activation of selected output units is shown when the evolving gestalt is probed with each role. The #s correspond to the number of phrases that have been presented to the network at that point. #1 means the network has seen "The teacher;" #2 means it has seen "The teacher ate;" etc. The activations (ranging between 0 and 1) are depicted as the darkened area of each box.

network has settled on an interpretation of the sentence. The thematic roles are represented with their appropriate fillers.

Specifics of the model

Input representation. Each sentence constituent can be thought of as a surface role/filler pair. It consists of one unit indicating the surface role of the constituent and one unit representing each word in the constituent. One unit stands for each of 13 verbs, 31 nouns, 4 prepositions, 3 adverbs, and 7 ambiguous words. Two of the ambiguous words have two verb meanings, three have two noun meanings, and two have a verb and a noun meaning. Six of the words are vague terms (e.g. someone,

The teacher ate the soup.

Sentence Gestalt Activations			
unit	#1	#2	#3
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			
17			
18			
19			
20			
21			
22			
23			
24			
25			

Role/Filler Activations	#1	#2	#3
agent			
person			
adult			
male			
female			
busdriver			
teacher			
action			
consumed			
ate			
gave			
threw(host)			
drove(motiv.)			
patient			
person			
adult			
child			
female			
schoolgirl			
thing			
food			
ball(party)			
steak			
soup			
crackers			

something, and food). For prepositional phrases, the preposition and the noun are both represented. For the verb constituent, the presence of the auxiliary verb "was" is likewise encoded by a separate unit. Articles are not represented, and nouns are assumed to be singular and definite throughout.

The surface role, location, of each constituent is coded by four units that represent location respective to the verb: pre-verbal, verbal, first-post-verbal, and n-post-verbal. The first-post-verbal unit is active for the constituent immediately following the verb, and the n-post-verbal unit is active for any constituent occurring after the first-post-verbal constituent. A number of constituents, therefore, may share the n-post-verbal position. For example, the sentence, "The ball was hit by someone with the bat in the park," would be encoded as the ordered set {pre-verbal/ball, verbal/(was, hit), first-post-verbal/(by, someone), n-post-verbal/(with, bat), n-post-verbal/(in, park)}.[2]

Output representation. The output has one unit for each of 9 possible thematic roles (e.g. agent, action, patient, instrument) and one unit for each of 45 concepts, including 28 noun concepts, 14 actions, and 3 adverbs. Additionally, there is a unit for the passive voice. Finally, there are 13 "feature" units, such as male, female, and adult. These units are included in the output to allow the demonstration of more subtle effects of constraints on interpretation (see Appendix A for the complete set of roles and concepts). This representation is not meant to be comprehensive. Instead, it is meant to provide a convenient way to train and demonstrate the processing abilities of the network. Any one role/filler pattern, then, consists of two parts. For the role, one of the 9 role units should be active, and for the filler, a unit representing the concept,

action, or adverb should be active. If relevant, some of the feature units or the passive voice unit should be active.[3]

Training environment. While the sentences often include ambiguous or vague words, the events are always specific and complete: each event consists of a specific action and each thematic role related to this action is filled by some specific concept. Accordingly, each event occurs in a particular location, and actions requiring an instrument always have a specific instrument.

Sentence/event pairs are created on-line during training from scaffoldings called sentence-frames. The sentence-frames specify which thematic roles and fillers can be used with that action. Each of the 14 actions has a separate sentence-frame. Four additional frames were made to cover passive versions of sentences involving the actions *kissed*, *shot*, *hit*, and *gave*.

To create a sentence/event pair, a sentence-frame is picked at random and then each thematic role is processed in turn. (Appendix B contains a sample sentence-frame.) For example, let's assume that the *Ate* sentence-frame is chosen. Agent is the first role processed. First, a concept to fill the role is selected from the set of concepts that can play the agent role in the *Ate* sentence-frame. This role/filler pair is added to the event description. Since some roles, such as instrument and location, may not be mentioned in the sentence, it is randomly determined, according to a preset probability, whether a role will be included in the sentence. If the role is to be included, a word is chosen to represent the filler in the sentence. Otherwise, the role is left out of the sentence, but it is still included in the event description. Since the agent role must be included in sentences about eating, it is placed in the sentence, and a word is chosen.

Assuming *busdriver* is chosen as the filler concept, a word to describe *busdriver* is selected. For example, the word "someone" might be chosen.

Next, the action role is processed. Since the *Ate* sentence-frame is being used, the action must be *ate*. A word to describe *ate* is then chosen: "consumed", for example. Then the patient is chosen. The probabilities of choosing particular patients depend upon what has been selected for the agent and action. Given the selection of *busdriver* as the agent, *steak* is a much more likely patient than *soup*. Let's assume that *steak* is selected, and that the word "steak" is chosen to represent it. In general, by changing the probabilities of selecting specific fillers as sentences are built, statistical regularities among the fillers will develop across the corpus.

In the same way, the remaining role/filler pairs for the sentence-frame are generated. Assuming only the first three roles are chosen to be included in this sentence, the input sentence will be, "Someone consumed the steak." The event will be the entire set of role/filler pairs {agent/busdriver, action/ate, patient/steak, instrument/knife, location/living-room, etc.}.

In this way, 120 different events can be generated with some being more likely to appear than others. The most frequent event occurs, on average, 5.5 times per 100 trials, but the least frequent event occurs only 9 times per 10,000. The number of words that can be chosen to describe an event and the option to include or eliminate optional constituents from the sentence brings the number of sentence/event pairs to 22,645.

The sentences are limited in complexity because of the limitations of the event representation. Only one filler can be assigned to a role in a particular sentence. Also, all the roles are assumed to belong to the sentence as a whole. Therefore, no embedded

clauses or phrases attached to single constituents are possible.

Training procedure details. After the processing of each constituent, the error produced by each role/filler pair is collected and propagated backward through the network (cf. Rumelhart, Hinton, & Williams, 1986). The weight changes from each sentence trial are added together and used to update the weights after every 60 trials.

The following values were used for the learning parameters. The learning rate, ε, was set to 0.0005, and momentum was set to 0.9. No attempt was made to optimize these values, so it is likely that learning time could be improved by tuning these parameters.

Results

Overall performance

First, we will assess the model's ability to comprehend sentences generally. Then we will examine the model's ability to fulfill our specific processing goals. Finally, we will examine the development of the model's performance across training trials.

When the model was able to process the passive sentences correctly, the simulation was stopped and evaluated. Correct processing was defined as activating the correct units more strongly than the incorrect units. After 330,000 sentence trials, the model began correctly processing the passive sentences in the corpus.

A set of 100 test sentence/event pairs was drawn randomly from the corpus. These sentence/event pairs were drawn without regard to their frequency during training, so seldom practiced pairs were as likely to appear in the test set as frequently practiced pairs. Of these pairs, 45 were set aside for separate analysis because they

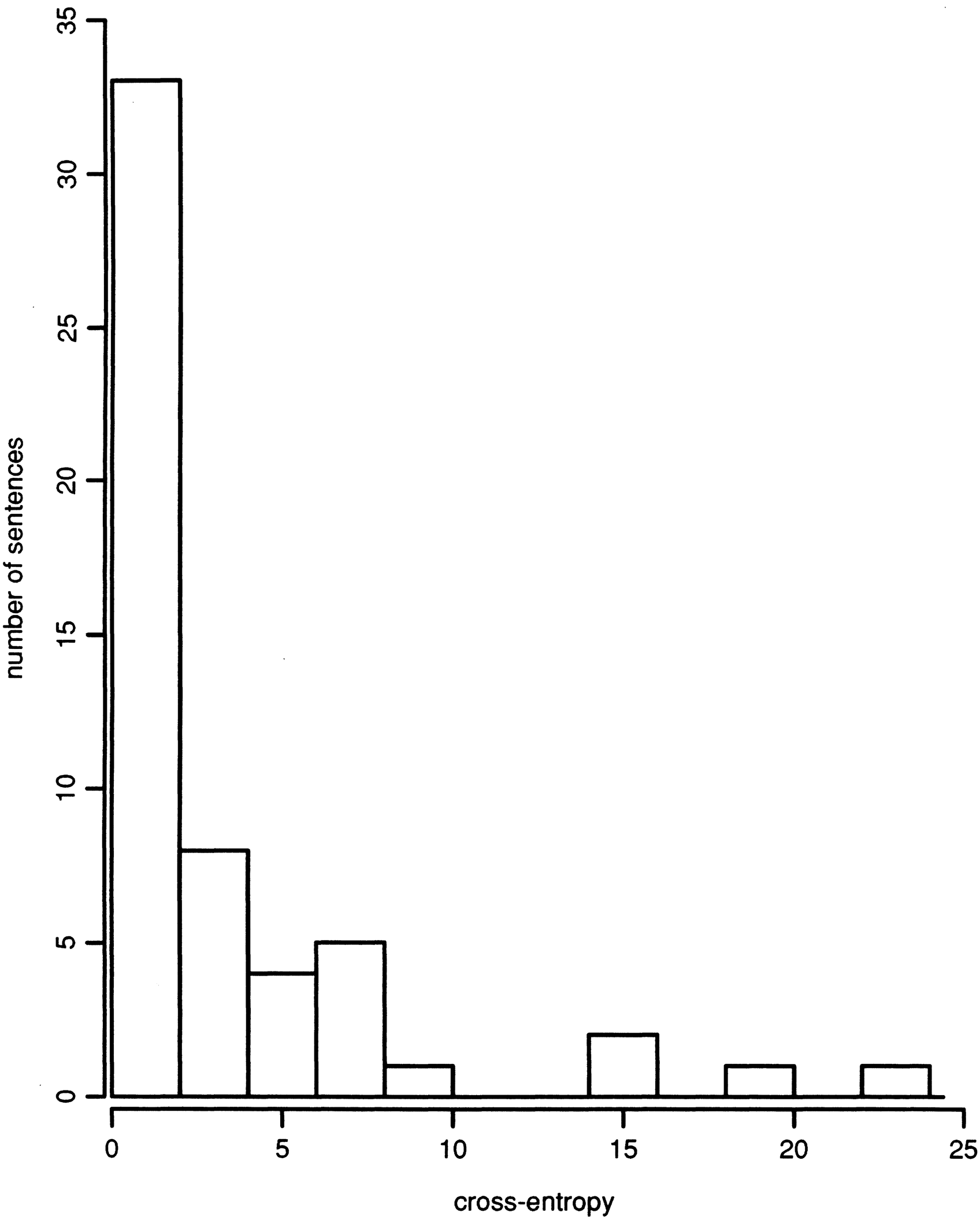
were ambiguous: at least two different interpretations could be derived from each. Of the remaining sentence/event pairs, every sentence contained at least one vague or ambiguous word, yet each had only one interpretation. These unambiguous sentence/event pairs were tested by first allowing the model to process all of the constituents of the sentence. Then the model was probed with each half of each constituent that was mentioned in the sentence. The output produced in response to each probe was compared to the target output. Figure 3 presents a histogram of the results.

Figure 3 - Histogram of unambiguous sentences

see following page

Figure 3. Histogram of the cross-entropy error for random sentences after 330,000 sentence trials. The sentences were drawn randomly from the corpus without regard to their frequency. A cross-entropy measure of between 0 and 10 results from sentences that are processed almost perfectly. Only small errors occur when an output unit should be completely activate (with a value of 1), but only obtains an activation of .7 or .8, or when a unit should have an activation of 0, but has an activation of .1 or .2. Cross-entropy errors of between 15 and 20 occur when one of the role/filler pairs is incorrect. For example, if teacher were supposed to be the agent, but the network activates busdriver, an error of about 15 would result.

Unambiguous Sentences



For these unambiguous sentences, the cross-entropy, summed over constituents, averaged 3.9 per sentence. Another measure of performance is the number of times an output unit that should be on is less active than an output unit that should be off. This situation occurred in 14 out of the 1710 possible cases, or on 0.8% of the opportunities.

The 14 errors were distributed over 8 of the 55 sentences. In five of the eight sentences, the error involved the incorrect instantiation of the specific concept or a feature of that concept referred to by a vague word. Two involved the incorrect activation of the concept representing a nonvague word. In each case, the incorrect concept was similar to the correct concept. Therefore, errors were not random; they involved the misactivation of a similar concept or the misactivation of a feature of a similar concept. The errors in the remaining sentence involved the incorrect assignment of thematic roles in a passive, reversible sentence: "Someone hit the pitcher" (see the section on learning for a discussion of this problem).

Additional practice, of course, improved the model's performance. Improvement is slow, however, because the sentences processed incorrectly are relatively rare. After a total of 630,000 trials, the number of sentences having a cross-entropy higher than 15 dropped from 3 to 1.

Performance on specific tasks

Our specific interest was to develop a processor that could correctly perform several important language comprehension tasks. Five typical sentences were drawn from the corpus to test each processing task. The categories and one example sentence for each are presented in Table 1.

Category	Example
Role assignment	
Active semantic	The schoolgirl stirred the kool-aid with a spoon.
Active syntactic	The busdriver gave the rose to the teacher.
Passive semantic	The ball was hit by the pitcher.
Passive syntactic	The busdriver was given the rose by the teacher.
Word ambiguity	The pitcher hit the bat with the bat.
Concept instantiation	The teacher kissed someone.
Role elaboration	The teacher ate the soup (with a spoon).

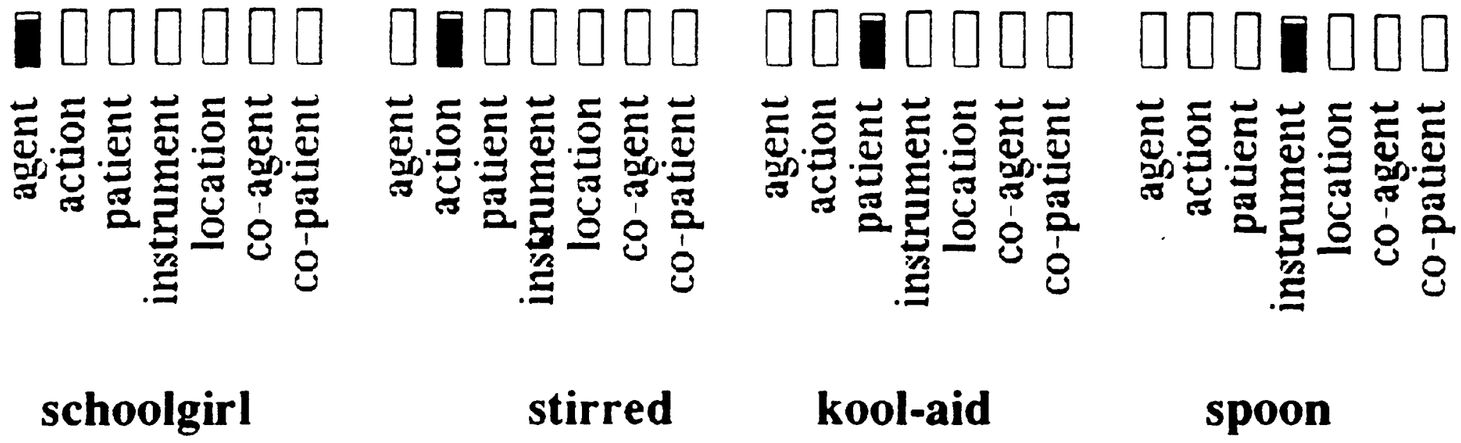
Table 1. The four categories of processing tasks and an example sentence of each. The parentheses denote the implicit, to be inferred, role.

The first category involves role assignment. The category was divided into four sub-categories based on the type of information available to help assign the correct thematic roles to constituents. Sentences in the active semantic group contain semantic information that can help assign roles. In the example from Table 1, of the concepts referred to in the sentence, only the schoolgirl can play the role of an agent of stirring. The network can therefore use that semantic information to assign schoolgirl to the agent role. Similarly, kool-aid is something that can be stirred, but cannot stir or be used to stir something else. After each sentence was processed, the sentence gestalt was probed with the filler half of each role/filler pair. The network then had to complete the pair by filling in the correct thematic role. For each pair, in each sentence, the unit representing the correct role was the most active. Sentences in the passive semantic category are processed equally well. Of course the semantic knowledge necessary to perform this task is never provided in the input or programmed into the network. Instead, it must be developed internally in the sentence gestalt as the network learns to process sentences. Syntactic information, though available, need not

be used in these cases; the semantic constraints suffice. In fact, if the surface location of the constituents is removed from the input, the roles are still assigned correctly.

To process sentences in the active and passive syntactic categories, however, the network cannot rely entirely on semantic constraints to assign thematic roles. To create this situation, pairs of reversible events were included in the training corpus, such as the

The schoolgirl stirred the kool-aid with a spoon.



The busdriver was given the rose by the teacher.

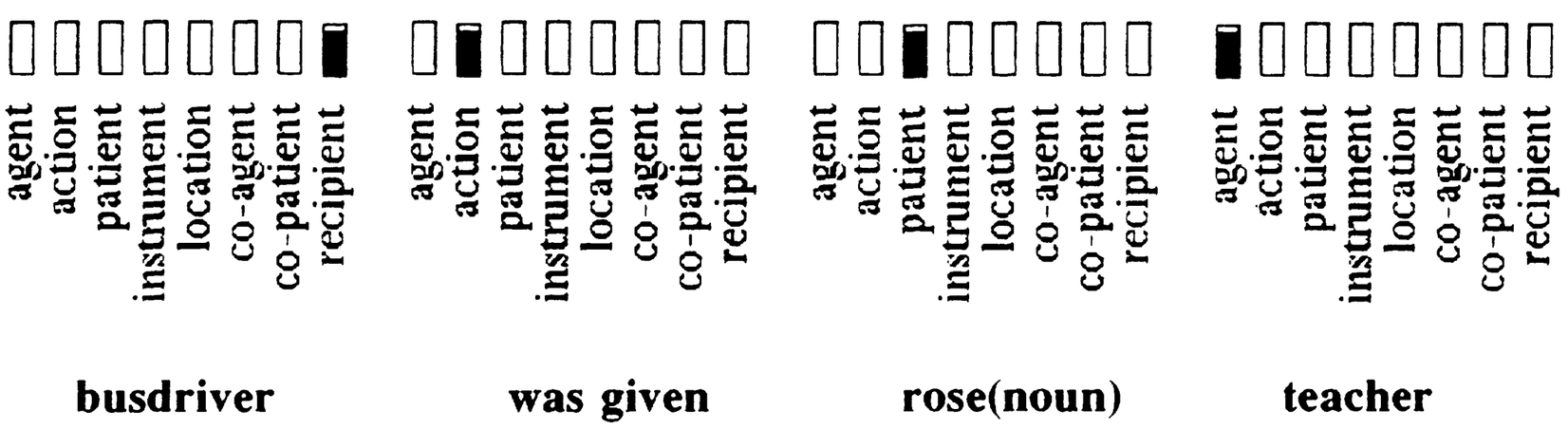


Figure 4. Role assignment. After a sentence is processed, the network is probed with the filler half of each role/filler pair. The activation over a subset of the thematic role units is displayed. The first sentence contains semantic information useful for role assignment, while the second sentence contains only syntactic information.

busdriver giving a rose to the teacher, and the teacher giving a rose to the busdriver. Both of these events were trained with equal frequency. Without a difference in frequency, there is no semantic regularity to help predict which of the two events a sentence refers to. The model must rely on syntactic information, such as word order, to assign the thematic roles. Passive sentences further complicate processing: the past participle and the "by" preposition provide cues designating the passive, but in themselves do not cue which person plays which role. The syntactic structure information must be used in conjunction with the passive cues to determine the correct role assignments. Again, for each role/filler pair in each test sentence, the correct role was the most active.

The remaining three categories involve the use of context to help specify the concepts referred to in a sentence. Sentences in the word ambiguity category contain one or more ambiguous words. After processing a sentence, the network was probed

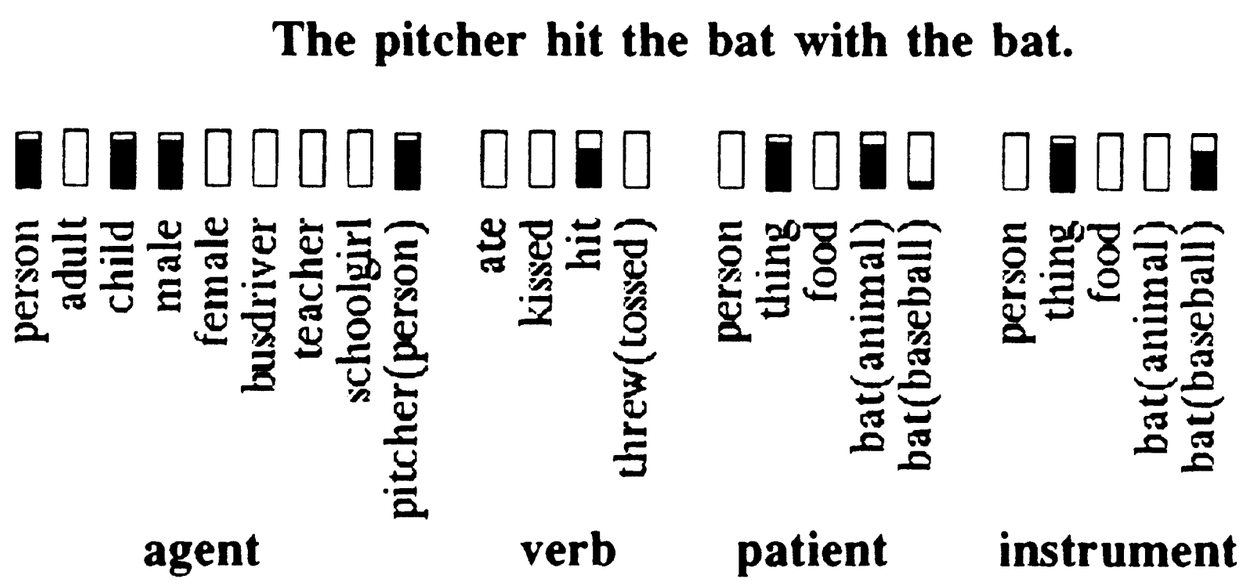


Figure 5. Word disambiguation. The sentence "The pitcher hit the bat with the bat" is processed by the network. The network is then probed with each thematic role in the event. The activation over a subset of the fillers is displayed. The network correctly disambiguates each word.

with the role half of each role/filler pair. The output pattern for the fillers were then examined. For all pairs in each test sentence, the correct filler was the most active.

Disambiguation requires the competition and cooperation of constraints from both the word and its context. While the word itself cues two different interpretations, the context fits only one. In "The pitcher hit the bat with the bat," pitcher cues both container and ball player. The context cues both ball player and busdriver because the model has seen sentences involving both people hitting bats. All the constraints supporting ball player combine, and together they win the competition for the interpretation of the sentence. As can be seen from the present example, even when several words of a sentence are ambiguous, the event which they support in common dominates the disparate events that they support individually. The processing of both instances of "bat" work similarly: the word and the context mutually support the correct interpretation. Consequently, the final interpretation of each word fits together into a globally consistent event.

Concept instantiation should work similarly. Though the word cues a number of more specific concepts, only one fits the context. Again, the constraints from the word and from the context combine to produce a unique, specific interpretation of the term. As with the disambiguation task, each test sentence was processed, and then the network was probed with the role half of each role/filler pair. The output filler patterns were examined to see if the correct concept and semantic features were instantiated. In each case, the correct concept and features were the most active.

Depending upon the sentence, however, the context may only partially constrain the interpretation. Such is the case in "The teacher kissed someone." "Someone" could refer to any of the four people found in the corpus. Since, in the network's experience,

females only kiss males, the context constrains the interpretation of "someone" to be either the busdriver or the pitcher, but no further. Consequently, the model can activate the *male* and *person* features of the patient while leaving the units representing *busdriver* and *pitcher* only partially active. The features *adult* and *child* are also

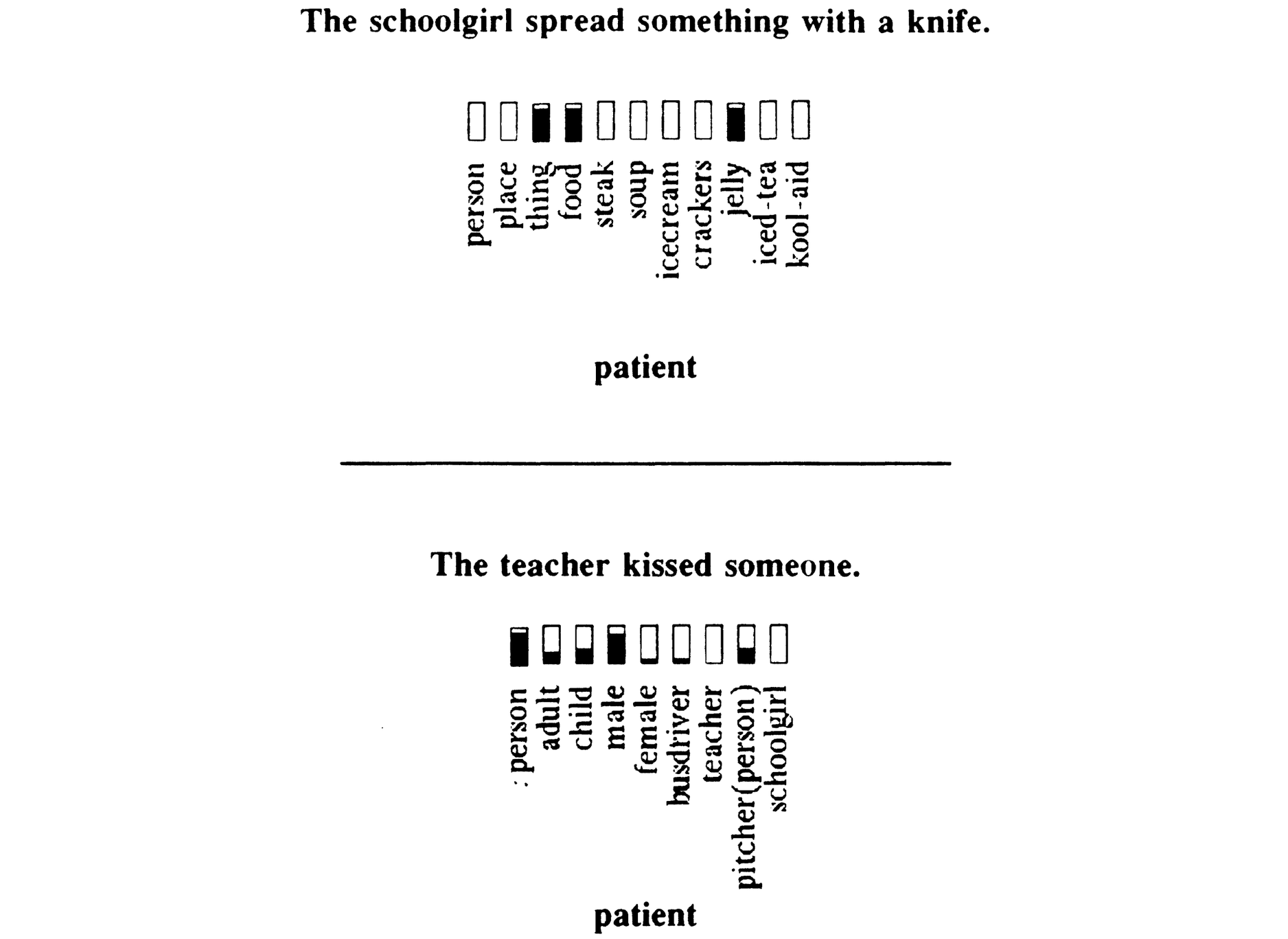


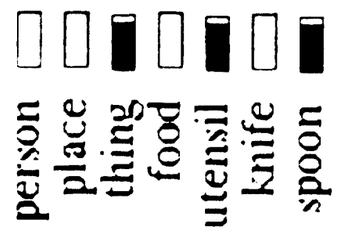
Figure 6. Concept instantiation. The network has learned that *jelly* is always the patient of *spread*. When the network processes "The schoolgirl spread something with a knife," it instantiates "something" as *jelly*. For the sentence "The teacher kissed someone," the network partially instantiates "someone" as a *male* and *person*, and either the *pitcher* or the *busdriver*.

partially and equally active because the busdriver is an adult while the pitcher is a child (see figure 6). While *pitcher* is slightly more active in this example, neither is activated above 0.5 (See the section on ambiguous sentences for an explanation of the difference in activations). In general, the model is capable of inferring as much information as the evidence permits: the more evidence, the more specific the inference. Word disambiguation can be seen as one type of this general inference process. The only difference is that for ambiguous words both the general concept and the specific features differ between the alternatives, while for vague words, the general concept is the same and only some of the specific features differ.

Finally, sentences in the role elaboration category test the model's ability to infer thematic roles not mentioned in the input sentence. For example, in "The teacher ate the soup," no instrument is mentioned, yet a spoon can be inferred. For each test sentence, after the sentence was processed, the network was probed with the role half of the to-be-inferred role/filler pair. The correct filler was the most active in each case. For role elaboration, the context alone provides the constraints for making the inference. Extra roles that are very likely will be inferred strongly. When the roles are less likely, or could be filled by more than one concept, they are only weakly inferred.

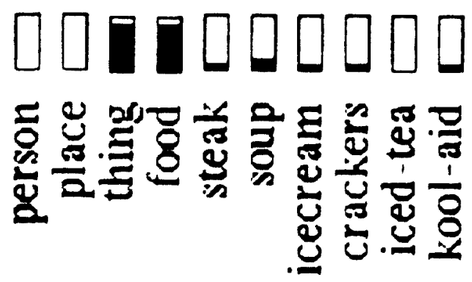
As it stands, there is nothing to keep the network from generalizing to infer extra roles for every sentence, even for events in which these roles make no sense. For instance, in "The busdriver drank the iced-tea," no instrument should be inferred, yet the network infers knife because of its association with the busdriver. It appears that since the busdriver uses a knife in many events about eating, the network generalizes to infer the knife as an instrument for his drinking. However, in events further removed

The teacher ate the soup.



instrument

The schoolgirl ate.



patient

Figure 7. Role elaboration. After processing the sentence "The teacher ate the soup," the network is probed with the instrument role. The filler activations are displayed. The network correctly infers *spoon*. For "The schoolgirl ate," the model must infer a patient. Because the schoolgirl is likely to eat a variety of foods, no particular food is well activated.

from eating, instruments are not inferred. For example, in "The busdriver rose," no instrument is activated. It appears, then, that generalization of roles is affected by the degree of similarity between events. When events are similar, elaborative roles may be generalized. When events are distinct, roles do not generalize, and the model has no reason to activate any particular filler for a role.

Immediate update

As each constituent is processed, the information it conveys modifies the sentence gestalt and strengthens the inferences it supports. But the beginning of a sentence may not always accurately predict its eventual full meaning. For example, in "The adult ate the steak with daintiness," the identity of the adult is initially unknown. After "The adult ate" has been processed, *busdriver* and *teacher* are equally active. After processing "The adult ate the steak," the model guesses that the agent is the *busdriver* since steak is typically eaten by busdrivers. Along with this inference, *gusto* is inferred as the manner of eating, since busdrivers eat with gusto. The model has, at this point, been led down the garden path toward an ultimately incorrect interpretation of the sentence. The next constituent processed, "with daintiness," only fits with the teacher and the schoolgirl. Since the sentence specifies an adult, the agent must be the *teacher*. The model must revise its representation of the event to fit with the new information by de-activating *busdriver* and activating *teacher*.

Figure 8 - Sentence Gestalt garden path

see following page

Figure 8. The sequential processing of a garden-path sentence. After "the steak" has been processed, the network instantiates "the adult" with the concept *busdriver*. When "with daintiness" is processed, network must reinterpret "the adult" to mean *teacher*.

The adult ate the steak with daintiness.

Sentence Gestalt Activations				
unit	#1	#2	#3	#4
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				
20				
21				
22				
23				
24				
25				
26				
27				
28				

Role/Filler Activations	#1	#2	#3	#4
agent				
person				
adult				
child				
male				
female				
busdriver				
teacher				
action				
ate				
shot				
drove(trans.)				
drove(motiv.)				
patient				
person				
adult				
child				
busdriver				
schoolgirl				
thing				
food				
steak				
soup				
crackers				
adverb				
gusto				
pleasure				
daintiness				

In general, as each constituent is processed, the information it explicitly conveys is added to the representation of the sentence along with implicit information implied by the constituent in the current situation. When the evidence is ambiguous and supports may conflicting inferences (such as after "The adult" has been processed) all the inferences are weakly activated in the sentence gestalt, and when new evidence suggests a different interpretation, the sentence gestalt is revised.

Ambiguous sentences

The ambiguous sentences in the test set were tested separately. As noted above, an ambiguous sentence has more than more consistent interpretation. For example, the adult in the sentence, "The adult drank the iced-tea in the living-room," can be instantiated with either *busdriver* or *teacher* as the agent, but the sentence offers no clues that *teacher* is the correct agent in this particular sentence/event pair in the test set. In these ambiguous cases the model should compromise and activate *busdriver* and *teacher* equally, causing two small errors. What the network typically did, however, was to activate one concept more than the other.

One reason for these differences in activations is the recent training history of the network. The sentence/event pairs trained more recently have a major impact on the weights and, therefore, on subsequent processing. Because selection of training examples occurs randomly, several sentences involving a particular agent may occur before a sentence/event involving a different agent is trained. Such training biases can lead to a bias in the activation of alternatives in ambiguous sentences. We tested this explanation by training the network on sentence/event pairs that consisted of an ambiguous sentence and the subordinate, weakly activated, event. From one to three

training trials were required to balance the activation of the subordinate event with that of the previously dominant event.

The network's sensitivity to aberrations is due to the dynamics of the activation function. Because the function is sigmoidal, it is sensitive to the value of the input in its middle range and insensitive at its extremes. Activation can be pegged on or off easily by using large positive or negative weights. The exact size of the weights are not critical. To obtain activations in the middle range, however, much finer calibration of the weights must be made. Consequently, recent changes to the weights can have a significant impact on performance in these subtle cases.

Learning

As the network learns to comprehend sentences correctly, a number of developmental phenomena can be observed. In fact, the only real failures in performance stem from a developmental effect. Problems in processing only arise in processing infrequent and irregular sentences. For example, sentences about the busdriver eating soup are rare. The network is seven times more likely to see a sentence about the busdriver eating steak than eating soup. This frequency difference creates a strong regularity between "The busdriver ate" and the concept *steak*. In a sentence about the busdriver eating soup, the word "soup" constrains the patient to be *soup*, while "The busdriver ate" partially constrains the patient to be *steak*. The constraints compete for an interpretation of the sentence. When the regularities are particularly strong, the contextual constraints can win the competition and cause the bottom-up activation from the word itself to be overridden.

Though this effect seems like a serious flaw, it is a flaw that the model shares

with people. In an illuminating experiment, Erickson and Mattson (1981) asked subjects questions like, "How many animals of each kind did Moses take on the Ark?" Subjects typically answered, "Two," despite their knowledge, when later asked, that Moses had nothing to do with the Ark. Constraints from the context overwhelmed the constraint from the word "Moses."

In the model, this frequency or regularity effect diminishes with training: the reliability of a constraint, its probability of correctly predicting the output, rather than its overall frequency, becomes increasingly important. The word "soup" perfectly predicts the concept *soup*: whenever "soup" appears in a sentence, the event contains the concept *soup*. On the other hand, the busdriver eats a variety of foods, so "the busdriver ate" is only 70% reliable as a predictor of *steak*. With increased training, even low frequency constraints are practiced. If they are reliable, they gain strength and eventually outweigh more frequent but less reliable constraints. Similar developmental trends occur as children learn language (MacWhinney, 1987). Progress is slow, but after a total of 630,000 trials, even these very infrequent and irregular sentences are processed correctly.

The early effect of frequency works for syntactic constraints as well as semantic constraints. As shown in Figure 9, the model masters sentences in the active voice sooner than it masters sentences in the passive voice. This difference is due to the greater frequency of sentences in the active voice in the corpus. While 14 sentence frames use the active voice, only 4 frames use the passive voice. After 330,000 trials, though, both voices are handled correctly.

The syntactic constraints develop more slowly than the regular semantic constraints. Yet while every sentence contains word order constraints, only an

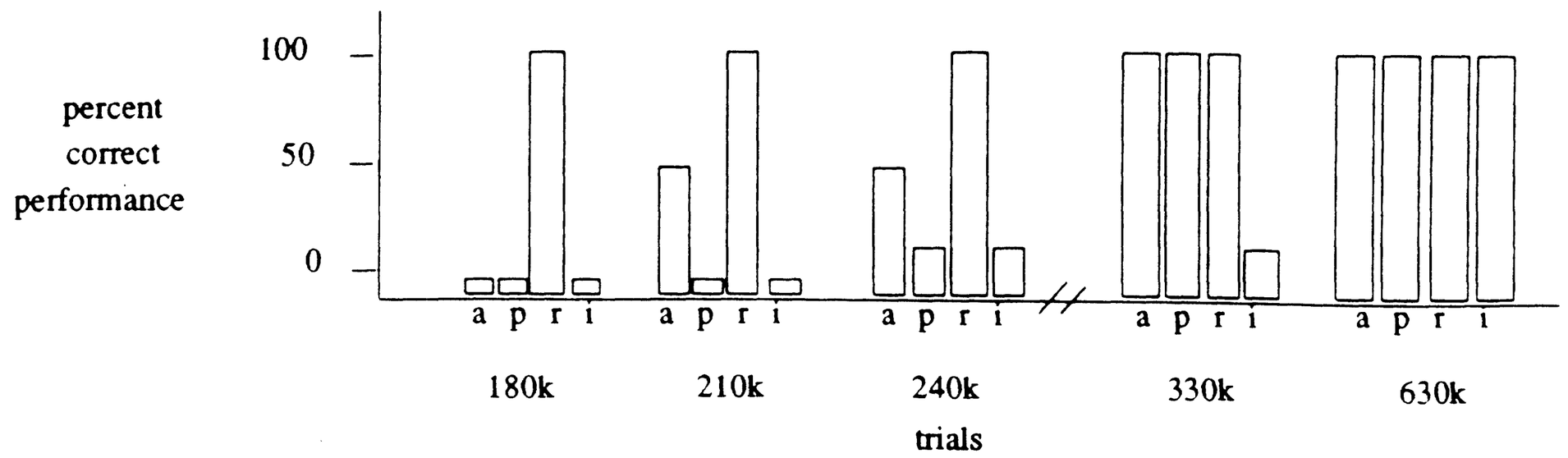


Figure 9. Development of performance. a - active syntactic (The busdriver kissed the teacher); p - passive syntactic (The teacher was kissed by the busdriver); r - regular semantic (The busdriver ate the steak); i - irregular semantic (The busdriver ate the soup). Correct performance means the correct concepts are more active than incorrect concepts.

occasional sentence will contain a particular semantic constraint. Based on the frequency of practice with particular constraints, then, the word order constraints should be learned much earlier than the semantic constraints. Two caveats to the frequency rule help explain this result. First, the syntactic constraints involve the conjunction of word order with the presence or absence of the passive markers, and such conjunctions are difficult to learn. Second, learning tends to generalize across semantically similar words, so training on one word can facilitate the learning of similar words.

Representations

While the input to the network is a local encoding, where each word is represented by a different unit, the network can create internal representations that are

distributed and that explicitly encode helpful semantic information. The weights running from the input layer to the first hidden layer can be seen as "constraint vectors" which determine how each word influences the evolution of the sentence gestalt. These constraint vectors are the model's bottom-up representation of each word. Words that impose similar constraints should develop similar constraint vectors. A cluster analysis of the weight vectors reveals their similarity. Separate cluster analyses was performed for unambiguous nouns and verbs (see figure 10).

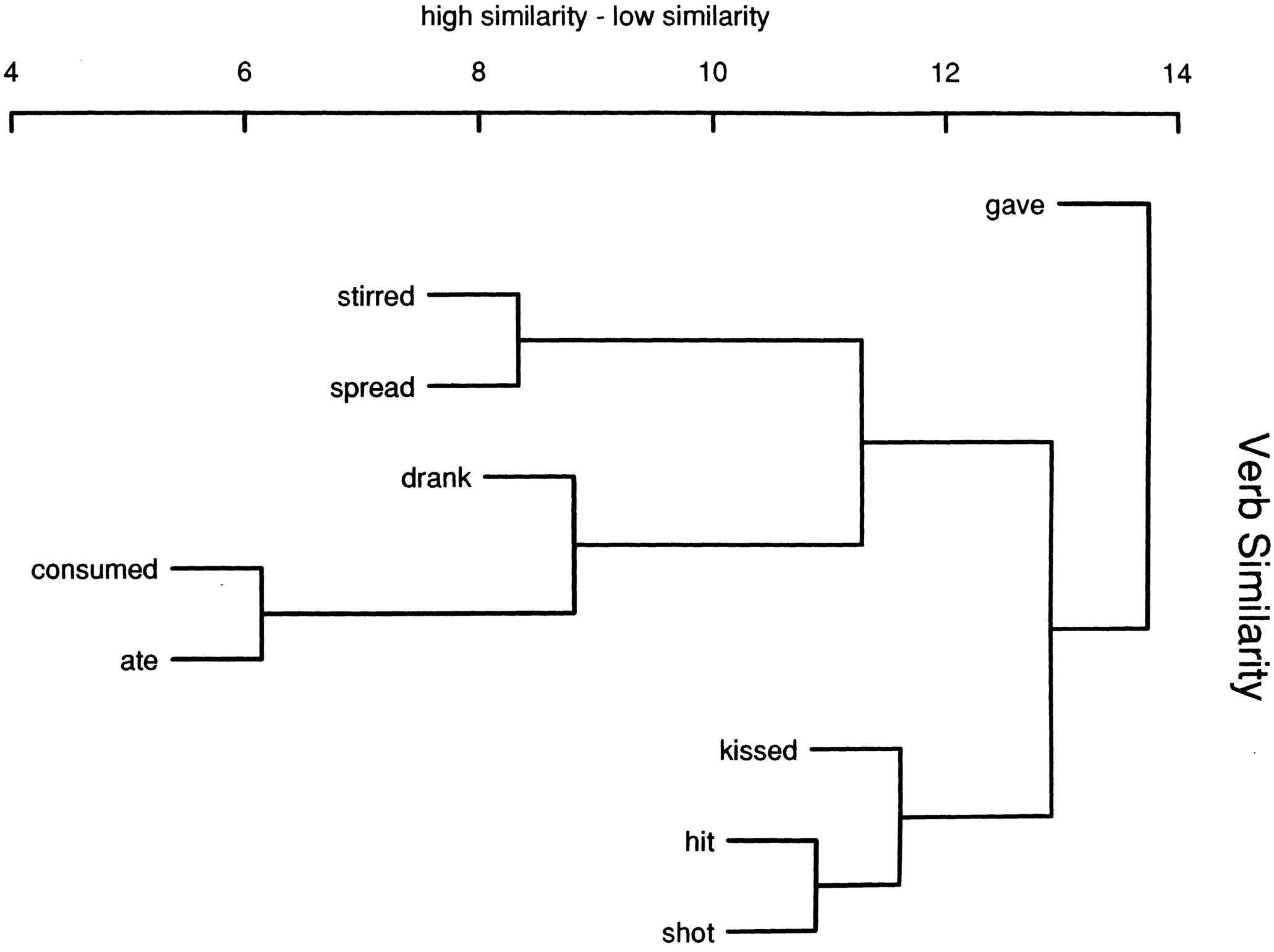
The verbs cluster into a number of hierarchical groups. One cluster contains the consumption verbs. Another contains stirred and spread. These two clusters then combine into a cluster involving people and food. Kissed, hit, and shot formed another cluster. For each of these verbs there were passive sentences in the corpus, and each could take an animate object. Gave, the only dative verb, stands apart from the other verbs. This clustering reflects the similarity of the case frames of the members of the

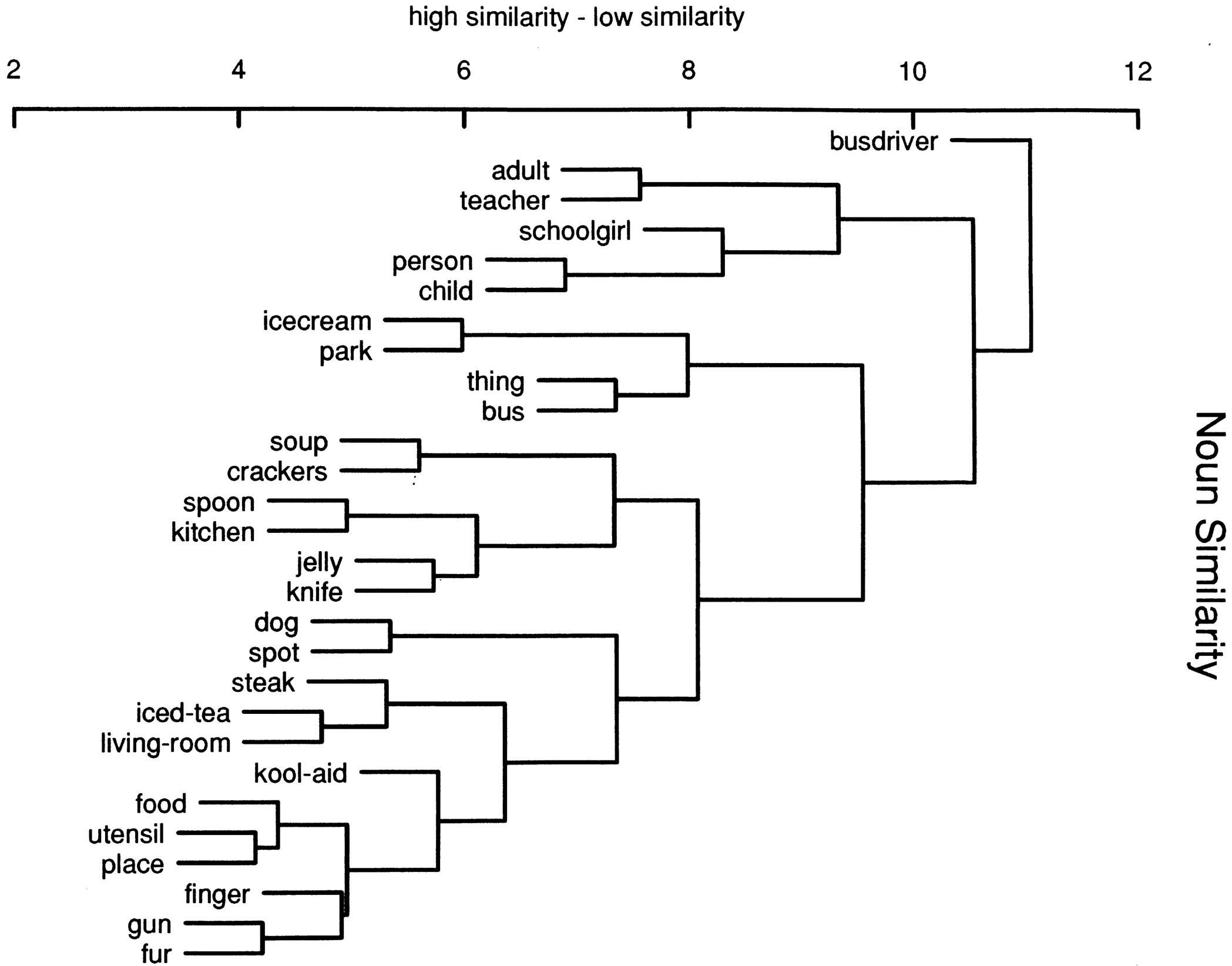
---

figure 10 - cluster analyses

see following pages

Figure 10. Cluster analyses. The analysis computes the similarity between the weight vectors leading from each input unit to the first hidden layer. The more similar two vectors or clusters of vectors, the sooner they are combined into a new cluster. Physical distance in the figure is irrelevant; only the clustering is important. For instance, stirred is not any more similar to drank than it is to ate.


high similarity - low similarity
4
6
8
10
12
14
gave
stirred
spread
drank
consumed
ate
kissed
hit
shot
Verb Similarity


high similarity - low similarity
2
4
6
8
10
12
busdriver
adult
teacher
schoolgirl
person
child
icecream
park
thing
bus
soup
crackers
spoon
kitchen
jelly
knife
dog
spot
steak
iced-tea
living-room
kool-aid
food
utensil
place
finger
gun
fur
Noun Similarity

different clusters.

The representations of nouns further reflect similarities in the constraints they impose on the evolving sentence gestalt. This similarity is reflected in two ways. As with the verbs, semantically similar words cluster: all of the people cluster, and "dog" and "spot" are very similar. Words that occur together in the same context also have similar representations. For example, ice cream clusters with park, and jelly clusters with knife. In the corpus, ice cream is always eaten in the park, and jelly is always spread with a knife. Consequently, their constraints on the output event-representation are similar. Their similar representations follow from the similar constraints they impose on the events described by the sentences.

Discussion

The SG model has been quite successful in meeting the goals that we set out for it, but it is of course far from being the final word on sentence comprehension. Here we briefly review the model's accomplishments. Following this review, we consider some of its limitations and how they might be addressed by further work.

Accomplishments of the model

One of the principle successes of the SG model is the fact that it correctly assigns constituents to thematic roles, based on syntactic and semantic constraints. The syntactic constraints are more difficult for the model to master than the semantic constraints, even though we have provided explicit cues to the syntax, in the form of the surface location of the constituents, in the input. The model does, however, come to master these constraints as they are exemplified in the corpus of training sentences.

Though syntactic constraints can be significantly more subtle than those our model has faced thus far, those it has faced are fairly difficult. To correctly handle active and passive sentences the model must map surface constituents onto different roles depending on the presence of various surface features elsewhere in the sentence.

The model also exhibits considerable capacity to use context to disambiguate meanings and to instantiate vague terms in contextually appropriate ways. Indeed, it is probably most appropriate to view the model as treating each constituent in a sentence as a clue or set of clues that constrains the overall event description, rather than as treating each constituent as a lexical item with a particular meaning. Although each clue may provide stronger constraints on some aspects of the event description than on others, it is simply not the case that the meaning associated with the part of the event designated by each constituent is conveyed by only that constituent itself.

The model likewise infers unspecified arguments roughly to the extent that they can be reliably predicted from the context. Here we see very clearly that constituents of an event description can be cued without being specifically designated by any constituent of the sentence. These inferences are graded to reflect the degree to which they are appropriate given the set of clues provided. The drawing of these inferences is also completely intrinsic to the basic comprehension process: no special separate inference processes must be spawned to make inferences, they simply occur implicitly as the constituents of the sentence are processed.

The model demonstrates the capacity to update its representation as each new constituent is encountered. Our demonstration of this aspect of the model's performance is somewhat informal; nevertheless, its capabilities seem impressive. As each constituent is encountered, the interpretation of all aspects of the event description

is subject to change. If we revert to thinking in terms of meanings of particular constituents, both prior and subsequent context can influence the interpretation of each constituent. Unlike most conventional sentence processing models, the ability to exploit subsequent context is again an intrinsic part of the process of interpreting each new constituent. There is no backtracking; rather, the representation of the sentence is simply updated to reflect the constraints imposed by each constituent as it is encountered.

The model learns to do all of the things we have described in the face of considerable ambiguity. Though the majority of sentences the model encounters have only a single interpretation, a substantial fraction of the sentences have more than one possible meaning. The model learns to interpret the unambiguous cases predominately correctly, albeit after a considerable amount of training. For the ambiguous cases, the model is sensitive to recent learning experiences. Instead of setting activations to actually match conditional probabilities, the interpretations of aspects of events that remain underspecified after all of the constituents have been encountered tend to vacillate based on recent, related training trials. Such vacillations are reminiscent of the frequent finding that humans generally do not notice ambiguity of sentences. Instead, they generally settle for one interpretation or the other, unless their attention is explicitly drawn to the ambiguity. The long-term, average probabilities of picking particular interpretations may reflect the statistical properties of the environment, while the moment to moment fluctuations reflect recent experience.

The gradual, incremental learning capabilities of the network underly its ability to solve the bootstrapping problem, that is, to learn simultaneously about both the mapping and meaning of constituents. The problem of learning mapping is central for

developmental psycholinguistics. Naigles, Gleitman, and Gleitman (1987) state that learning the mapping between words and concepts based only on statistical information seems impossible because, "at a minimum, it would require such extensive storage and manipulation of contingently categorized event/conversation pairs as to be unrealistic." Yet it is exactly by using such information that our model solves the problem. The model learns the syntax and semantics of the training corpus simultaneously. Across training trials, the model gradually learns which aspects of the event description each constituent of the input constrains and in what ways it constrains these aspects.

The problem of discovering which event in the world a sentence describes when multiple events are present would be handled in a similar way, though we have not modeled it. Again, the aspects of the world that the sentence actually describes would be discovered gradually over repeated trials, while those aspects that spuriously co-occur with these described aspects would wash out. For both the bootstrapping and the ambiguous reference problem, then, our model takes a gradual, statistical approach. We do not want to overstate the case here, since the child learning a language confronts a considerably more complex version of these problems than our model does. Our sentences are pre-segmented into constituents, are very simple in structure, and are much fewer in number than the sentences a child would hear. However, the results demonstrate that the bootstrapping and ambiguous reference problems might ultimately be overcome by an extension of the present approach.

Many of the accomplishments of the SG model are shared by predecessors. Cottrell (1985), Cottrell and Small (1983), Waltz and Pollack (1985), and McClelland and Kawamoto (1986) have all demonstrated the use of syntactic and semantic constraints in role assignment and meaning disambiguation. Of these, the first two

embodied the immediate update principle, but did not learn, while the third learned in a limited way, and had a fixed set of input slots.

The greater learning capability of our model allows it to find connection strengths that solve the constraints embodied in the corpus, without requiring the modeler to induce these constraints and trying to build them in by hand. It also allows the model to construct its own representations in the sentence gestalt, and this ability allows these representations to be considerably more compact than in other cases.

Some previous models have used conjunctive representations in which role/filler pairs are explicitly represented by units pre-assigned to represent either specific role/filler pairs (as in Cottrell, 1985; Cottrell & Small, 1983; Waltz & Pollack, 1985) or particular combinations of role features and filler features (McClelland & Kawamoto, 1986). Particularly when such representations are extended so that triples, rather than simply pairs, can be represented (St. John & McClelland, 1987; Touretzky & Geva, 1987), these networks can become intractably large even with small vocabularies. The present model avoids intractable size by learning to use its representational capacity sparingly to represent just those role/filler pairings that are consistent with its experience. This ability prevents the model from being able to represent totally arbitrary events: its representational capacities are strongly constrained by the range of its experience. In this regard the model seems similar to humans: it is widely known that human comprehension is strongly influenced by experience (Bartlett, 1932; Chase & Simon, 1973).

Deficiencies and limitations of the model

The model has several limitations and a few obvious deficiencies. The model

only addresses a limited number of language phenomena. It does not address quantification, reference and co-reference, coordinate constructions, or many other phenomena. Perhaps the most important limitation is the limitation on the complexity of the sentences, and of the events that they describe, that the model can process. In general, it is necessary to characterize the surface roles and fillers of sentences with respect to their superordinate constituents. Similarly in complex events, there may be more than one actor, each performing an action in a different sub-event of the overall event or action. Representing these structures requires head/role/filler triples instead of simple role/filler pairs.

One solution is to train the model using triples rather than pairs as the sentence and event constituents. The difficulty lies in specifying the non-sentence members of the triples. These non-sentence members would stand for entire structures. Thus they would be very much like the patterns that we are currently using as sentence gestalts. It would be desirable to have the learning procedure induce these representations, but this is a bootstrapping problem that we have not yet attempted to solve.

Another limitation of the model is the use of local representations both for concepts and for roles. The present model used local representations of concept meanings only for convenience; in reality we would suppose that the conceptual representations underlying events would be represented by distributed patterns (Hinton, McClelland, & Rumelhart, 1986). This kind of representation would have several advantages. Context has the capability not only of selecting among highly distinct meanings such as those of flying bats and baseball bats, but also, we believe, of shading meanings, emphasizing certain features and altering properties slightly as a function of context (McClelland & Kawamoto, 1986). Both of these phenomena are easily

captured if we view the representation of a concept as a distributed pattern.

Similarly, there are several problems with the concept of role which are solved if distributed representations are used. It is often difficult to determine whether two roles are the same, and it is very difficult to decide exactly how many different roles there are. If roles were represented as distributed patterns, these issues would simply fall by the wayside. In earlier work (McClelland & Kawamoto, 1986), it was necessary to invent distributed representations for concepts, but recently a number of researchers have shown that such representations can be learned (Hinton, 1986; Miikkulainen & Dyer, 1988; Rumelhart, semantic reps:). The procedure should also apply to distributed representations of roles.

Another limitation of the model is the explicit presence in the input of surface role markings. We had originally hoped that the network would not need such explicit markings, but would come to represent, in the SG, information about position in the surface parse. While we have been able to get the network to learn without such markings, learning time increases dramatically. It may be that the task we have imposed on the SG, to represent the entire event as soon as possible, conflicts with maintaining information about position in the string. In a different task, where the network must attempt to anticipate the next input, there have been several demonstrations that networks can learn to keep track of parse position, at least for small finite-state grammars (Elman, 1988; Cleeremans and Servan-Shreiber, personal communication).

A final limitation is the small size of the corpus used in training the model. Given the length of time required for training, one might be somewhat pessimistic about the possibility that a network of this kind could master a substantial corpus.

However, it should be noted that the extent to which learning time grows with corpus size is extremely hard to predict for connectionist models, and is highly problem dependent. For some problems (e.g. parity), learning time per pattern increases more than linearly with the number of training patterns (Tesauro, 1987), while for other problems (e.g. negation), learning time per pattern actually can decrease as the number of patterns increases. Where the current problem falls on this continuum is not yet know.

Learning time per pattern is closely related to generalization in connectionist networks. One limitation of our experiments on the SG model, though not necessarily a limitation of the model itself, is that we have not really assessed generalization. As things stand, a reader might be tempted to suppose that our model has simply memorized the corpus, and could not generalize at all to sentences containing novel uses of words. However, our analysis of the input representations produced by the words used in our simulations suggests that generalization might be possible. These analyses demonstrate that the network learns to assign input representations to words that reflect the constraints they impose on the event description. It seems likely that a considerable part of the specification of these constraints might be derivable by the network from experience on a subset of the possible contexts where a word can occur. The interpretation acquired in these experiences would then cause the new word to behave like other similar words in contexts in which it was not trained. Illustrations that back propagation networks can generalize in this way are provided by Hinton (1986), Taraban, McDonald and MacWhinney (In Press), and Rumelhart (1987). We are not at present in a position to say just how well the SG model would do in this regard.

One final deficiency of the model is its tendency to activate fillers for roles that do not apply to a particular frame. This tendency could perhaps be overcome by explicit training that there should be no output for a particular role, but this seems inelegant and impractical, especially if we are correct in believing that the set of roles is open-ended. The absence of roles seems somehow implicit in events, rather than explicitly noted. Perhaps event representations that preserved more detail of the real-world event would provide the relevant implicit constraints.

Conclusion

The SG model represents another step in what will surely be a long series of explorations of connectionist models of language processing. The model is an advance in our view, but there is still a very long way to go. The next step is to find ways to extend the approach to more complex structures and more extensive corpora, while increasing the rate of learning.

References

Anderson, R. C., & Ortony, A. (1975). On putting apples into bottles: A problem of polysemy. *Cognitive Psychology, 7,* 167-180.

Bartlett, F. C. (1932). *Remembering: An experimental and social study.* Cambridge: Cambridge University Press.

Carpenter, P. A. & Just, M. A. (1977). Reading comprehension as the eyes see it. In M. A. Just & P. A. Carpenter (Eds.), *Cognitive processes in comprehension.* Hillsdale, NJ: Erlbaum.

Chase, W. G. and Simon, H. A. (1973). Perception in chess. *Cognitive Psychology, 4,* 55-81.

Cleeremans, A. and Servan-Schreiber, D. Personal communication. Department of Psychology, Carnegie-Mellon University.

Cottrell, G. W. (1985). A connectionist approach to word sense disambiguation. Dissertation, Computer Science Department, University of Rochester, NY.

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

van Dijk, T. A. & Kintsch, W. (1983). *Strategies of discourse comprehension.* Orlando, Florida: Academic Press.

Elman, J. L. (1988). Finding structure in time. CRL technical report 8801. Center for Research in Language, University of California, San Diego, CA.

Erickson, T. D. & Mattson, M. E. (1981). From words to meaning: A semantic illusion. *Journal of Verbal Learning and Verbal Behavior, 20,* 540-551.

Fillmore, C. J. (1968). The case for case. In E. Bach & R. T. Harms (Eds.), *Universals in linguistic theory.* New York: Holt, Rinehart, & Winston.

Gleitman, L. R. & Wanner, E. (1982). Language acquisition: The state of the state of the art. In E. Wanner & L. R. Gleitman (Eds.), *Language acquisition: The state of the art.* Cambridge, MA: Cambridge University Press.

Hinton, G. E. (1981). Implementing semantic networks in parallel hardware. In G. E. Hinton and J. A. Anderson (Eds.), *Parallel models of associative memory.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Hinton, G. E. (1986). Learning distributed representations of concepts. Paper presented to the 8th Annual Conference of the Cognitive Science Society. Amherst, MA.

Hinton, G. E. (1987). Connectionist learning procedures. Technical report #CMU-CS-87-115. Department of Computer Science, Carnegie-Mellon University.

Hinton, G. E., McClelland, J. L., and Rumelhart, D. E. (1986). Distributed representations. In D. E. Rumelhart, J. L. McClelland, and the PDP Research Group (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition, Volume 1.* Cambridge, MA.: MIT Press.

Jordan, M. I. (1986). Attractor dynamics and parallelism in a connectionist sequential machine. Paper presented to the 8th Annual Conference of the Cognitive Science Society. Amherst, MA.

MacWhinney, B. (1987). Competition. In B. MacWhinney (Ed.), *Mechanisms of language acquisition: The 20th annual Carnegie symposium on cognition.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Marcus, M. P. (1980). *A theory of syntactic recognition for natural language.* Cambridge, MA: MIT Press.

Marslen-Wilson, W. & Tyler, L. K. (1980). The temporal structure of spoken language understanding. *Cognition, 8,* 1-71.

McClelland, J. L. & Elman, J. L. (1986). Interactive processes in speech perception: The TRACE model. In J. L. McClelland, D. E. Rumelhart, and the PDP Research Group (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition, Volume 2.* Cambridge, MA.: MIT Press.

McClelland, J. L. & Kawamoto, A. H. (1986). Mechanisms of sentence processing: Assigning roles to constituents. In J. L. McClelland, D. E. Rumelhart, and the PDP Research Group (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition, Volume 2.* Cambridge, MA.: MIT Press.

McClelland, J. L. & Rumelhart, D. E. (1981). An interactive activation model of context effects in letter perception: Part 1. An account of basic findings. *Psychological Review, 88,* 375-407.

McKoon, G. & Ratcliff, R. (1981). The comprehension processes and memory structures involved in instrumental inference. *Journal of Verbal Learning and Verbal Behavior*, 20, 671-682.

Miikkulainen, R. and Dyer, M. G. (1988). Building distributed representations without microfeatures. Technical report. Artificial Intelligence Laboratory, Computer Science Department, University of California, Los Angeles, CA.

Naigles, L. G., Gleitman, H., & Gleitman, L. R. (1987). Syntactic bootstrapping in verb acquisition: Evidence from comprehension. Department of Psychology, University of Pennsylvania.

Quine, W. V. (1960). *Word and Object.* Cambridge, MA: Harvard Press.

Rumelhart, D. E., (1987). Colloquium presented to the Department of Computer Science, Carnegie-Mellon University.

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

St. John, M. F. & McClelland, J. L. (1987). Reconstructive memory for sentences: A PDP approach. Paper presented to the Ohio University Inference Conference, *Proceedings Inference: OUIC 86.* University of Ohio, Athens, OH.

Taraban, R., McDonald, J., and MacWhinney, B. (In Press). Category learning in a connectionist model: Learning to decline the German definite article. In R. Corrigan (Ed.), Milwaukee Conference on Categorization. Philadelphia: John Benjamins.

Tesauro, G. (1987). Scaling relationships in back-propagation learning: Dependence on training set size. Technical report. Center for Complex Systems Research, University of Illinois at Urbana-Champaign, Champaign, Il.

Touretzky, D. S. and Geva, S. (1987). A distributed connectionist representation for concept structures. Paper presented to the 9th Annual Conference of the Cognitive Science Society. Seattle, WA.

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

Appendix A
Input and Output Representations

Input

surface locations:
pre-verbal, verbal, post-verbal-1, post-verbal-n

words:
consumed, ate, drank, stirred, spread, kissed, gave, hit, shot, threw, drove, shed, rose
someone, adult, child, dog, busdriver, teacher, schoolgirl, pitcher, spot
something, food, steak, soup, ice cream, crackers, jelly, iced-tea, kool-aid
utensil, spoon, knife, finger, gun
place, kitchen, living-room, park, bat, ball, bus, fur
gusto, pleasure, daintiness
with, in, to, by
was

Output

roles:
agent, action, patient, instrument, co-agent, co-patient, location, adverb, recipient

concepts:
ate, drank, stirred, spread, kissed, gave, hit, shot, threw(tossed), threw(hosted), drove(trans), drove(motiv), shed(verb), rose(verb)
busdriver, teacher, schoolgirl, pitcher(person), spot
steak, soup, ice cream, crackers, jelly, iced-tea, kool-aid
spoon, knife, finger, gun
kitchen, living-room, shed(noun), park
rose(noun), bat(animal), bat(baseball), ball(sphere), ball(party), bus, pitcher(container), fur
gusto, pleasure, daintiness

noun features:
person, adult, child, dog, male, female
thing, food, utensil
place, in-doors, out-doors

verb features:
consumed, passive

Appendix B
Sample Sentence-Frame

Hit

agent 100[1]
25[2] busdriver 70[3] adult 20 person 10
verb 100
100 hit 100
patient 100
25 shed-n 80 something 20
instrument 50
100 bus 80 something 20 with[4]
40 ball-s 80 something 20
location 50
100 park 100 in
instrument 50
100 bat-b 80 something 20 with
10 bat-a 80 something 20
location 50
100 shed-n 100 in
instrument 50
100 bat-b 80 something 20 with
25 pitcher-p 70 child 20 person 10
location 50
100 park 100 in
instrument 50
100 ball-s 80 something 20 with
25 teacher 70 adult 20 person 10
verb 100
100 hit 100
patient 100
34 pitcher-c 80 something 20
location 50
100 kitchen 100 in
instrument 50
100 spoon 80 something 20 with
33 pitcher-p 70 child 20 person 10
location 50
100 living-room 100 in
instrument 50
100 pitcher-c 80 something 20 with
33 schoolgirl 70 child 20 person 10
location 50
100 kitchen 100 in
instrument 50
100 spoon 80 something 20 with
25 pitcher-p 70 child 20 person 10

verb 100
100 hit 100
patient 100
40 ball-s 80 something 20
location 50
100 park 100 in
instrument 50
100 bat-b 80 something 20 with
10 bat-a 80 something 20
location 50
100 shed-n 100 in
instrument 50
100 bat-b 80 something 20 with
25 bus 80 something 20
location 50
100 park 100 in
instrument 50
100 ball-s 80 something 20 with
25 busdriver 70 adult 20 person 10
location 50
100 park 100 in
instrument 50
100 ball-s 80 something 20 with
25 schoolgirl 70 child 20 person 10
verb 100
100 hit 100
patient 100
34 pitcher-c 80 something 20
location 50
100 kitchen 100 in
instrument 50
100 spoon 80 something 20 with
33 spot 80 dog 20
location 50
100 kitchen 100 in
instrument 50
100 spoon 80 something 20 with
33 teacher 70 adult 20 person 10
location 50
100 kitchen 100 in
instrument 50
100 spoon 80 something 20 with

[1]Include a role in the input the this probability.
[2]Choose this filler with this probability.
[3]Choose this word with this probability.
[4]The word appears in this prepositional phrase.

Footnotes

[1]The situation, as defined in the learning procedure, is the combination of the previous sentence gestalt, the current constituent, and the current probe. It would be desirable to define the situation solely in terms of the sequence of sentence constituents. While our results suggest that the sentence gestalt learns to save all the relevant information from earlier constituents, we have no proof that it does.

[2]We originally tried presenting the constituents without their surface location, hoping the network would transform the temporal order of the constituents into a spatial pattern and then use the spatial pattern to produce syntactic constraints, like word order, to help interpret the sentence. Through simulation, we have noted that the network can learn this process, though only with great difficulty.

[3]A second output layer was included in the simulations. This layer reproduced the sentence constituent that fit with the role/filler pair being probed. Consequently, the model was required to retain the specific words in the sentence as well as their meaning. Since this aspect of the processing does not fit into the context of the current discussion, these units are not discussed further.