PARSNIP: A Connectionist Network that Learns Natural Language Grammar from Exposure to Natural Language Sentences

Stephen José Hanson

and

Judy Kegl

Bell Communications Research
435 South Street
Morristown, NJ 07960

Princeton University
Cognitive Science Laboratory
221 Nassau Street
Princeton, NJ 08542

Abstract

Linguists have pointed out that exposure to language is probably not sufficient for a general, domain-independent, learning mechanism to acquire natural language grammar. This "poverty of the stimulus" argument has prompted linguists to invoke a large innate component in language acquisition as well as to discourage views of a general learning device (GLD) for language acquisition. We describe a connectionist *non-supervised learning* model (PARSNIP[1]) that "learns" on the basis of exposure to natural language sentences from a million word machine-readable text corpus (Brown corpus). PARSNIP, an *auto-associator*, was shown three separate samples consisting of 10, 100 or 1000 syntactically tagged sentences, each 15 words or less. The network learned to produce correct syntactic category labels corresponding to each position of the sentence originally presented to it, and it was able to generalize to another 1000 sentences which were distinct from all three training samples. PARSNIP does sentence completion on sentence fragments, prefers syntactically correct sentences, and also recognizes novel sentence patterns absent from the presented corpus. One interesting parallel between PARSNIP and human language users is the fact that PARSNIP correctly reproduces test sentences reflecting one level deep center-embedded patterns which it has never seen before while failing to reproduce multiply center-embedded patterns.

Keyword Topics: Connectionist Models, Neural Nets, Learning, Language Acquisition

---

1. The name PARSNIP was chosen to emphasize that the present model is not a parser, but a "snippet" or precursor to a parser and is most similar to a syntactic analyzer. Our work was supported in part by a grant to Princeton University from the James S. McDonnell Foundation. We would like to thank Donald Walker, Stu Feldman and the connectionist group at Bellcore for comments on previous versions of this paper.

Introduction

Connectionist approaches to language processing (Feldman, 1985; Rumelhart & McClelland, 1986) have recently gained attention because of a need to simultaneously integrate diverse sources of information about the syntax, semantics, and pragmatics of a sentence. One important aspect of these neural-like models is their ability to combine information from various sources while at the same time allowing these sources to mutually constrain each other, reducing the need to prioritize one type of information over another during parsing.

Many questions arise concerning the computational nature of connectionist models and their potential role in natural language processing. Central to connectionist models is a *learning* process which determines how structure and rule governed behavior emerges. Unfortunately, the learning rules so far proposed (Ackley, Hinton & Sejnowski, 1985; Rumelhart, Hinton & Williams, 1986) focus primarily on the frequency of occurrence of relevant structural units within a given domain and require explicit supervision over the recognition and coding of generalizations concerning each stimulus encountered. Such constraints on learning procedures raise serious questions about the possibility of modeling natural language learning in a connectionist framework. Studies in learnability theory (Chomsky, 1957) have shown that natural language syntax cannot possibly be induced from the first-order statistics (e.g., transition matrices or conditional probabilities) available through exposure to an infinite number of examples. Children acquiring natural language are sensitive to a set of universal constraints on structural configurations and relations in language such as the A over A Condition, Subjacency, etc. (see Radford, 1981 for a description of these constraints). Even though violations of these conditions have not been explicitly corrected, discouraged, nor even experienced; children avoid violating these conditions even in their earliest linguistic utterances (Chomsky, 1965; Randall, 1982).

The acquisition of natural language poses particular problems for any learning approach. Language acquisition cannot rely on any explicit information about the grammaticality, usage, frequency, possible constituency, or any structural information about the sentence other than the linear order and cooccurrence of words in the sentence--and that information is hindered by performance errors, incomplete sentences and general noise. Under such conditions, it is hard to imagine how syntax, and natural language generally, is acquired at all. Chomsky (1972) approached this problem by assuming a nativist perspective in which the child was seen as using incoming language data in conjunction with innate linguistic knowledge to formulate hypotheses about possible grammatical rules and constraints.

Previous Work

Other computational models of language acquisition from both connectionist and rule-based approaches have tended to assume that a large amount of previous structure must be present to learn natural language syntax. A recent model (Berwick, 1985) incorporating linguistic assumptions from a Government and Binding perspective (Chomsky, 1981 and subsequent work) uses a "repair" operation on syntactic rules that are already present but need to be tuned properly. This tuning is based on incremental positive evidence in that sentences the learner hears are assumed to be grammatical and each new sentence must be incrementally accounted for. This type of acquisition where positive evidence and reactionary generalization is enforced is sometimes referred to the the "subset principle" (Berwick, 1985), Interestingly, the

connectionist model proposed here can be seen as consistent with the subset principle.

Connectionist models (Feldman, 1985; Rumelhart & McClelland, 1986; Selman, 1985; McClelland & Kawamoto, 1986) have tended to provide the system with explicit rules, syntactic structure or both. They have allowed the network to learn the proper conditions under which to apply these rules or to recognize specific relations between constituent structures.

Explicit analysis of the kinds of preconditions or structure needed prior to learning natural language grammar have yet to be considered for a connectionist model. For example, no connectionist models currently exist which build up their syntactic knowledge from mere exposure to positive examples and the subsequent incremental addition of new sentences. This model begins with no assumptions about syntactic structure nor any special expectations about properties of syntactic categories other than the fact that they exist.

The Present Model

We begin with the assumption that natural language reveals to the hearer a rich set of linguistic constraints and that observable syntactic regularities serve to delimit the possible grammars that can be learned. We are not making an anti-nativist argument, in fact, the present model actually contributes to the analysis of the tradeoff between innate syntactic knowledge and previously unrecognized syntactic regularities in the data that could be used to induce grammar. Connectionist models which learn in this way can offer a new paradigm for nativist research. By filtering out those data which can be learned, we may delineate those aspects of the knowledge of language which are truly hardwired.

PARSNIP uses a variation of a *backpropagation* technique (Rumelhart, Hinton & Williams, 1986) called "auto-association" which was originally proposed by Rumelhart and Hinton. These models are multi-layer learning networks (MLL; Hanson & Burr, 1987) that have units associated with input and output as well as a modifiable set of intermediate units called "hidden units." Although backpropagation is strictly a supervised technique, auto-association is not. The difference lies in the teacher signal. Backpropagation requires a separate teacher signal for every input-output pair, whereas auto-association uses the input as the teacher signal. The auto-association network's task is to produce a veridical copy of the input with which it is presented. It must recognize this input as something it has seen before.

This seemingly straightforward task becomes difficult when the network is exposed to a large number of stimuli or when the the number of "hidden units" is small compared to the number of input/output units, forcing a compression or reduction of the information which is encoded during learning. Reducing the number of encoding units is likely to yield a new (compressed) encoding of the input information in order to adequately map it to the output, akin to chunking smaller units into higher order constituents. The auto-associator may extract regularities more general than those exhibited by the input stimuli, or it may discover features or complexes of features that are useful in predicting the output stimulus.

We are asking the following question of our network: Can it induce grammar-like behavior (rule-governed behavior) from simple exposure to a large corpus of natural language sentences. Several specific questions will also be posed: After learning on a specific set of input, can the network generalize to sentences never seen before? Does it prefer sentences that are syntactically correct? Can it recognize sentences that are more complex than those that would be predicted by simple conditional probabilities on the combinations of fragments it has

previously seen? And finally, after learning on a sizeable natural language corpus, is the network resistant to learning sentences which violate syntactic well-formedness conditions purported to be universally applicable. This last question is a particularly interesting one, and is indicative of the types of questions that should be posed. If PARSNIP does not recognize such sentences or resists learning such sentences (not in this paper) after nothing more than exposure to data, this would lead us to suspect that rather than being an innate property of the learner, these constraints and conditions follow directly from regularities in the data.

A key aspect of grammar induction is the ability of the network to recognize forms that are syntactically correct but did not appear in training. Concurrently, it must not recognize syntactically incorrect forms that also never appeared in the training sample. This is a differential generalization constraint. Not only must the network generalize to new sentences, it must have a means of determining grammaticality; and worse yet, it must do so strictly on the basis of positive evidence.

Input Representation and Stimuli

PARSNIP was exposed to sentences from the Brown corpus (Francis & Kučera, 1979) consisting of one million words of running text. This corpus, compiled over a 10 year period, is composed of 500 text samples each consisting of approximately 2000 words. The texts are representative of 6 separate categories and approximately 19 subcategories, including newspaper text, religious books, technical books and novels.

President Kennedy today pushed aside other White House business to devote all his time and attention to working on the Berlin crisis address he will deliver tomorrow night to the American people over nationwide television and radio.

n-tl np nr vbd rb ap jj-tl nn-tl nn to vb abn pp$ nn cc nn in vbg in at np nn nn pps md vb nr nn in at jj nns in jj nn cc nn .

My advice , if you live long enough to continue your vocation , is that the next time you're attracted by the exotic , pass it up -- it's nothing but a headache. As you can count on me to do the same. Compassionately yours , S. J. Perelman

pp$ nn , cs ppss vb jj qlp to vb pp$ nn , bez cs at ap nn ppss+ber vbn in at jj , vb ppo rp -- pps+bez pn cc at nn . cs ppss md vb in ppo to do at ap . rb pp$$

She was a living doll and no mistake -- the blue-black bang , the wide cheekbones , olive-flushed , that betrayed the Cherokee strain in her Midwestern lineage , and the mouth whose only fault , in the novelist's carping phrase , was that the lower lip was a trifle too voluptuous.

pps bedz at vbg nn cc at nn -- at jj nn , at jj nns , jj , wps vbd at np nn in pp$ jj-tl nn , cc at nn wp$ ap nn , in at nn$ vbg nn , bedz cs at jjr nn bedz at nn ql jj .

Figure 1: Example Sentences Taken From the Brown Corpus

We chose the Brown corpus because it is one of the few sample corpora where each word of text is associated with a tag which indicates its syntactic category. The tags for each individual word

were determined by linguistically informed judges. Examples of text are shown in Figure 1. The three sentences are taken from the beginning, middle and end of the corpus and provides some idea of the diversity of sentence types and topics.

Below each sentence is a string of syntactic tags. There are approximately 81 unique word class tags comprised from about 6 kinds of syntactic information including major form classes ("parts of speech"), function words, inflectional morphs and punctuation. Tags were also combined during the labeling process in order to create new codes where needed. This compounding resulted in a total of 467 unique syntactic codes over the entire corpus. We used a nine bit binary representation to code all 467 categories then input these binary representations to the auto-associator. Tags were assigned to bit pattern codes by frequency of occurrence in the corpus; most frequent were assigned to most active input codes while least frequent were assigned to less active input codes[2] (sparser). To restrict sentence diversity, the length of the sentences shown to the network was limited to 15 words or less.[3] The Brown Corpus contains approximately 35,000 sentences of 15 words or less.

Architecture

The PARSNIP network consist of a total of 585 units and 24,615 connections. Each unit's Fan Out is completely connected to units above. The unit Fan In was combined by a linear integration function over the activation states below it and over the weights connected to these states. The unit Fan Out was normalized over the interval zero to one and was compressed in the high and low ends of the scale. This type of function (e.g., logistic) transforms activation at a unit to something like "firing rate" for a neural interpretation, or "likelihood" if a probabilistic interpretation is given (Hanson & Burr, 1987).

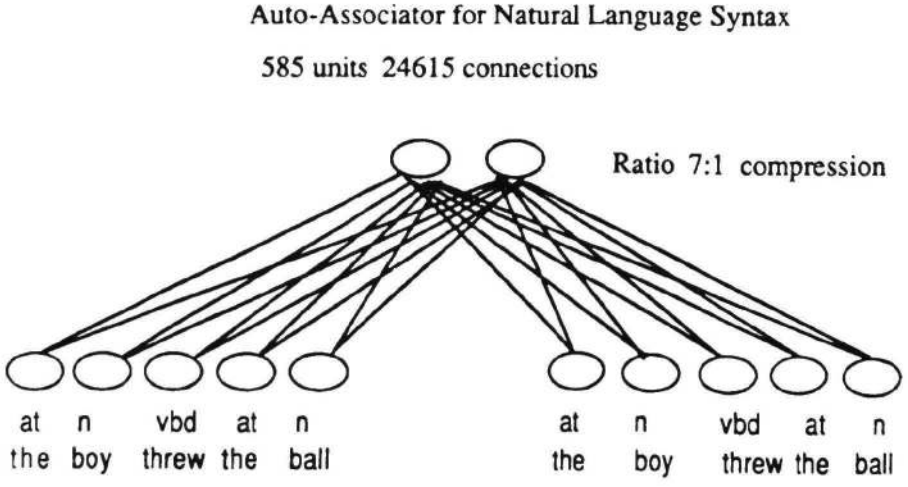

Figure 2: Schematic Version of Auto-Associator

2. Experiments were also attempted with random code assignments and there seemed to be little difference in the learning; although generalization performance has yet to be compared.

3. One consequence of restricting the length of sentences was the elimination of sentences containing relative clauses from the corpus. The absence of these sentence types will prove important in subsequent discussions.

A schematic version of the architecture of the auto-associator is shown in Figure 2. The input included 270 units coding 15 word positions (not including punctuation) and 14 word boundary codes. The output was identical to the input, also consisting of 270 positions. Hidden units varied in number from 10 to 60, although the data reported here is for 45 hidden units. This limitation on the number of hidden units provides a 7 to 1 compression of the data through the hidden layer.

The learning procedure was implemented with the generalized delta rule (Rumelhart, Hinton, Williams, 1986) at the output layer and was applied recursively to the layer below (between the hidden units and the input). The targets for the output layer were the input values themselves. The weights were adjusted by the following formula:

$$\Delta w_{ij}^{n+1} = \eta\,(o_{jp} * \delta_{ip}) + \alpha\,\Delta w_{ij}^{n} \qquad (1)$$

The parameter η represents the rate at which any particular sample error can affect the weights. α is a parameter that determines the effect that past deltas have had on the present delta. For α equal to 1, the present weight change and past weight change have the same effect in the weight update. An o_{jp} is the value for the jth unit and the pth pattern. And δ is the error gradient for the ith unit and the pth pattern. All experiments used an η of .1 and an α of .3.

PARSNIP Experiments

Sentences including punctuation (e.g., periods) were entered into one side of the auto-associator with padding (effectively zero or no input) after the period in order to uniformly fill 15 positions. Starting with random weights, a forward activation on the input produced activation on the output, also in 15 nine bit positions. The nine bit patterns were then compared to the input bit pattern, yielding the errors for each output value. These errors were then used to adjust the weights as specified in the delta rule.

In three separate training sessions, the PARSNIP network was separately trained[4] on three distinct sets of of sentences of sizes 10, 100 and eventually 1000.

4. The auto-associator/back-propagation simulator was written for a vectorizing FORTRAN compiler on a Convex C1 computer. Simulation runs, dependent on problem size, took any where from 5 hours to 3 1/2 weeks.

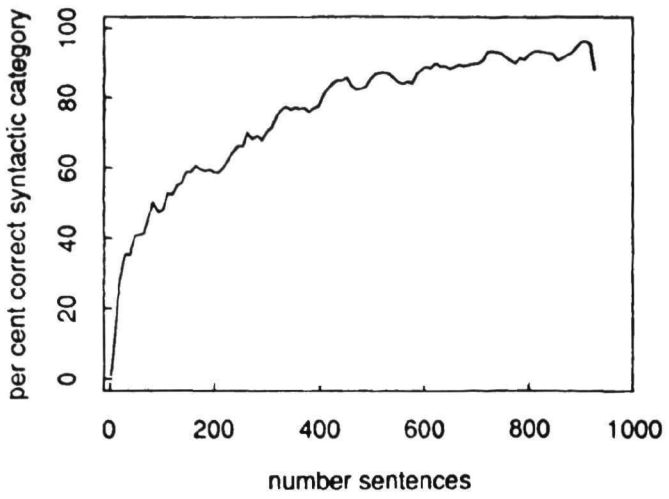

Figure 3: Learning of 10 Sentences from the Brown Corpus

The network was exposed to each set until criterion was reached (>95% correct on the entire set) or until no positive slope in the learning curve was detected. Errors were calculated from the number of missed *syntactic categories*. Thus a single bit error in the nine bit code would be counted as a miss of the entire category. Figure 3 shows percent correct ((1-error)*100) for the 10 sentence set as a function of the number of sentence presentations. Criterion was reached after about 100 cycles through the 10 sentences, namely, after about 1000 sentence presentations. In Figure 4 we show the transfer point to a new set of 100 sentences after having learned on the 10 sentence set. The first point in this graph (Figure 4) shows the last point from the 10 sentence set (Figure 3) and the next point shows the network's performance on a new sentence. Notice that performance drops dramatically from about 97% correct to 50% correct.

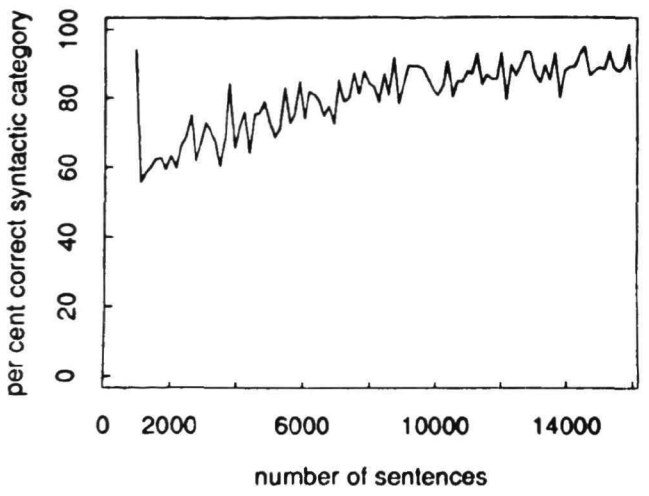

Figure 4: Learning of 100 Sentences from the Brown Corpus

Recall that the presence of word boundary information and end of sentence punctuation will allow the network to get at least 50% correct if it is able to retain just this information. In this case, after the learning on the 10 sentence set, word boundary information is all the network seems able to retain (also see below).

The network reaches criterion after about 160 cycles on the 100 sentences, namely, in about 16,000 sentence presentations. Notice that the learning curve is much more jagged in this

case as compared to the 10 sentence set. Apparently, the learning of some sentence structures tends to compete with the learning of other sentence structures[5]. Finally, Figure 5 indicates the beginning of transfer to 1000 new sentences. The first point, as before, is the last point of the learning curve for the 100 sentences, and the next point shows the response of the network to a new sentence. Again, the drop is rapid. But, this time slightly more information is retained about sentence structure as can be seen by the fact that the drop only reaches about 60%. As learning proceeds, it follows a gentle positive slope, although the jaggedness of the learning curve is much greater, and learning criterion is never reached with this sentence set. To ensure that the asymptote was reached, the sentences were cycled through 180 times (180,000 sentence presentations). This time it became apparent that the network had difficulty encoding all 1000 sentences. The final performance level achieved exhibited correct recognition on about 85% of the sentences.

Acquisition by Trials. The initial output of the network involves codes that are associated with low activation. That is, the network is inhibitory in early stages of acquisition. This is attributable to the fact that error reduction drives weight changes and to the sparseness of codes. For example, if most of the codes which the network is exposed to are sparse, that is have few 1's in the target, then the network can significantly reduce error by turning off output bits and thereby making the network inhibitory. This produces a tendency for the network to retrieve codes associated with low activation. Because of the sorting of codes by frequency, these will also be low frequency categories.

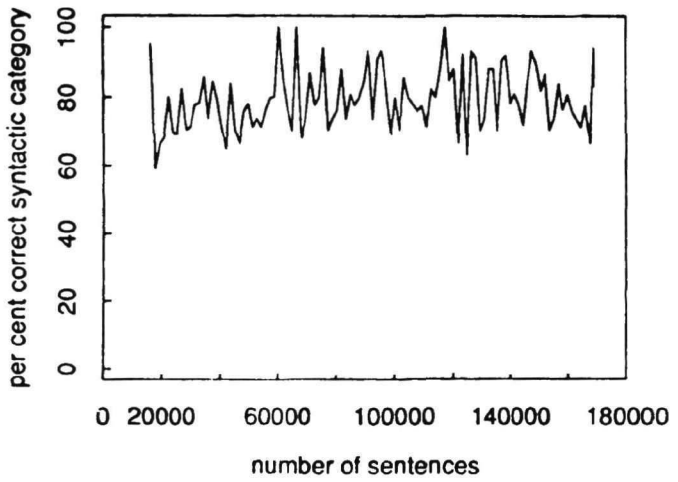

Figure 5: Learning of 1000 Sentences From the Brown Corpus

Within the first 30 trials (sentences), the network seems to pick up the first obvious regularity, that of word boundaries. Next, within the next 100 or so sentences, mass nouns and personal pronouns begin to be correctly predicted, as well as a few two sequence syntactic codes like

5. This type of learning curve is characteristic of learning rates that are too high for the sample. It is possible that too few hidden units are present for optimal learning. To control for the the possibility that the learning rate was too high it was dropped to half its value (.05). However, a similar amount of jaggedness was still apparent in the learning curve. In addition, experiments where 1/3 more hidden units were added in conjunction with smaller learning rates did not result in a substantial change in the texture of the curve.

article+noun and preposition+noun. As learning precedes, more complex forms begin to appear, but not with obvious predictability. Further analysis where sequences of tags are tracked through learning trials should be revealing.

Generalization Performance

The weights for all three earlier sample sizes were retained for a generalization test. With learning turned off ($\eta = 0$, $\alpha = 0$), each network was shown 1000 new sentences which were distinct from the 1110 sentences the three networks originally learned. The percent correct on the 1000 new sentences was recorded and the results are shown in Figure 6. On the x axis is the size of the sample of sentences the network had previously learned, and on the y axis is the percent correct of sentences which the network was able to predict from a novel set of 1000 sentences. Notice that the function is increasing, that is, more prior training on sentences produces greater generalization to novel sentences. As previously described, knowledge about the 10 sentence network drops to word boundary knowledge, losing about 50% of its sentence knowledge. The 100 sentence network retains about 10% more information about sentence structure (syntactic category relations), losing about 40% of what it had learned. Finally, the 1000 sentence network seems to be generalizing at about the same rate (84%) at which it had asymptotically learned. As these are log-log coordinates, it appears there is a hint that the 3 points approximate a power function of prior learning on sentence sample size.

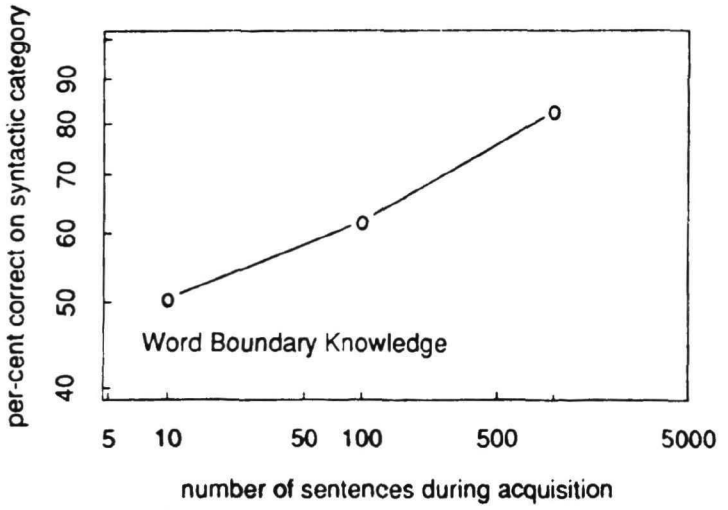

Figure 6: Generalization Performance

Recognition Performance

PARSNIP's main task is to recognize a sentence either as one it has seen before or as one that it might have seen before. This is the type of performance one might expect from any associative memory in which a large number of patterns have been stored. However, PARSNIP is much more than a pattern storer. It is in fact able to behave in a rule governed way with respect to sentence completion and recognition of sentence types it has never seen before. The question is whether the composition of sentence fragments or constituents are determined on the basis of first order statistics (conditional probabilities between sentence fragments) or whether they can be attributed to more complex generalizations arising from exposure to a large number of sentence types. All the remaining experiments were performed using the 1000 sentence network.

Pattern Completion. The first task that PARSNIP was asked to perform was sentence completion on the basis of partial input. In Figure 7 we show a sample interaction with PARSNIP. The syntactic tags representing the sentence *the boy threw a ball*, are clamped on one side of the input, Then, PARSNIP produces the same sentence, i.e. "article noun past-tense verb article noun". Suppose now that the verb is left out of the sentence and, instead, PARSNIP is shown an ambiguous code at the third position in the sentence. In this case, PARSNIP produces on the output side in the third position the tag "past-tense verb," or as a second guess "verb". That is, generalizations it has made concerning the possible structure of sentences cause PARSNIP to be reminded of the syntactic categories that best fit the empty slot in the sentence it was shown.

INPUT: ARTICLE NOUN P-VERB ARTICLE NOUN (The boy threw the ball)
PARSNIP: ARTICLE NOUN P-VERB ARTICLE NOUN

INPUT: ARTICLE NOUN <BLANK> ARTICLE NOUN
PARSNIP: ARTICLE NOUN <VERB P-VERB> ARTICLE NOUN

Figure 7: Sample Interaction with PARSNIP: Pattern Completion

Disambiguation. Another task that PARSNIP was asked to perform was one of syntactic disambiguation given a set of possible syntactic codes for a lexical item appearing in a particular sentence position. This task was similar to pattern completion except that the network was given a number of items and asked to produce the correct one. In Figure 8 we show a sample interaction where PARSNIP is given the sentence *The horse raced past the barn fell.* The word "past" could appear in a sentence as either an adverb, a preposition an adjective or a noun. The word "past" in this sentence begins the prepositional phrase "past the barn".

INPUT: ARTICLE NOUN P-VERB PREPOSITION ARTICLE NOUN P-VERB
PARSNIP: ARTICLE NOUN P-VERB PREPOSITION ARTICLE NOUN P-VERB

INPUT: ARTICLE NOUN P-VERB *ADVERB* ARTICLE NOUN P-VERB
PARSNIP: ARTICLE NOUN P-VERB *PREPOSITION* ARTICLE NOUN P-VERB

Figure 8: Sample Interaction with PARSNIP: Disambiguation

A dramatic way to demonstrate disambiguation in PARSNIP is to clamp the *incorrect* syntactic choice (adverb instead of preposition) as shown in figure 8. In response to this deliberate introduction of misinformation, PARSNIP *edits* the sentence and actually inserts the correct syntactic category. In this case, it should be noted that this particular sentence never appears in the 1000 sentence corpus to which PARSNIP was exposed.

Recursion. Sentence embedding, the ability of grammar to produce a sentence within another sentence, is considered a characteristic defining feature of natural languages. Paradoxically, it has also been shown that sentence recursion is not an unlimited feature of

natural language processing. Even at the next level of embedding (a sentence within a sentence within a sentence), human language users have difficulty (Miller, 1962). Therefore, general recursive rules must be filtered out somehow, and usually memory constraints are invoked in order to do this.

```
INPUT:    ARTICLE NOUN ARTICLE NOUN P-VERB P-VERB
PARSNIP:  ARTICLE NOUN ARTICLE NOUN P-VERB P-VERB

INPUT:    ARTICLE NOUN ARTICLE NOUN ARTICLE NOUN P-VERB P-VERB P-VERB
PARSNIP:  ARTICLE NOUN ARTICLE NOUN ARTICLE NOUN P-VERB NOUN VERB
```

Figure 9: Sample Interaction with PARSNIP: Recursion

In the sample interaction in Figure 9, PARSNIP is able to recognize the sentence *The rat the cat chased died.* This recognition occurs despite the lack of even a single occurrence of a center-embedded sentence within the corpus. Nonetheless, PARSNIP is able to respond to this sentence as something it recognizes. Apparently, PARSNIP is able to bind together constituents that have been used in other contexts. However, when a doubly embedded sentence, e.g., *the rat the cat the dog bit chased died* is clamped on the input side, PARSNIP produces a partial sentence but does not recognize this second level of recursion. Although constituents similar to those found in single level center-embedding are available in this more complex center-embedded, the the failure might be seen in terms of the number of and distance between constituents that must be bounded by PARSNIP's recognition rule. In other words, PARSNIP is not able to recognize constituents that it has previously recognized because they are bounded by constituents that may be unfamiliar or have not previously been useful in syntactic prediction. Note also that this effect is also independent of any memory constraints since PARSNIP is exposed to a total sentence in parallel.

Adjacency Constraints. In English, a direct object must be adjacent to a verb in order to receive case from it and thereby be allowed (licensed) to occur in object position. (This statement is phrased within the terminology of a Government-Binding approach (Chomsky, 1981).) English speaking children will probably seldom or never hear *John gave quickly the book*, where quickly intervenes between a verb and its object and blocks the assignment of accusative case by virtue of destroying the adjacency relation between the verb (the case assigner) and the direct object (the NP which must receive case). Furthermore, children acquiring English will never be explicitly discouraged from using these sentences if they should happen to hear them, e.g., "By the way, don't say this sentence". A key question in the evaluation of how language is acquired concerns the ability of the network to avoid generalizing to sentences that have adjacency violations of this type and which are not present in the training set.

INPUT: NOUN ADVERB VERB ARTICLE NOUN (men quickly steal the food)
PARSNIP: NOUN ADVERB VERB ARTICLE NOUN

INPUT: NOUN VERB ADVERB ARTICLE NOUN (men steal quickly the food)
PARSNIP: NOUN ADVERB WAS ARTICLE NOUN

Figure 10: Sample Interaction with PARSNIP: Adjacency Constraints

In Figure 10 we show PARSNIP failing to recognize an adjacency violation, in fact, it actually attempts to move what was the VERB (now retireved as "WAS") closer to the direct object. Note that PARSNIP can also recognize *men quickly steal* or *men steal quickly* implying that the presence of the direct object is critical for this recognition failure.

Discussion

Auto-association is clearly not a plausible model for language acquisition. That is, repeated parallel exposure to a sentence with enforced production of that sentence is not a reasonable cognitive model of language acquisition, nor of a language learner's grammar production. Perhaps the closest parallel to PARSNIP's situation is that of a learner engaged in the abstract intensive study of sentences and sentence structure (similar to the activities of a linguist). It is also important to note that unlike human language learners, the network has no sense of temporal order. For PARSNIP sentences have no beginning, middle or end, but rather they exist as patterns which can be used to account for the structures it encounters. Nonetheless, there are some important parallels between the task given to PARSNIP and the task that arises for children as they learn natural language. Both PARSNIP and the child are only exposed to sentences from natural language, they both must induce general rules and larger constituents from just the regularities to which they are exposed, both one the basis of only positive evidence.

PARSNIP's ability to generalize from what it has learned to new sentences indicates that some general knowledge of constituent structure has been extracted from its experience with natural language sentences. A significant amount of coverage of sentence types occurs after training on 1000 as compared to the original 10 sentences.

It is far more interesting to us to have discovered that PARSNIP can differentially generalize to sentences that can appear in natural language (center embeddings) but cannot recognize sentences which violate natural language constraints (multiple center embeddings). As evidence that PARSNIP is using rule-like representations or possibly possesses something comparable to a grammar, we feel it is important to point out the fact no center embedded sentences appeared in the training set. In fact, even the number of adjoined relative clauses was almost nil as a result of the limitation on sentence length. Apparently, constraints from the sentences already learned allows PARSNIP to differentially generalize as though syntactic rules are in operation.

Further, the constituents that PARSNIP chooses tend neither to be predictable from first order statistics nor to be able to be generated from simple finite state grammars. PARSNIP prefers sequences of syntactic categories that often are the least likely to be predicted on the basis of the frequency with which one category follows the other in the corpus. For example, in one pattern completion interaction when PARSNIP was given the phrase *the destruction of the*

city <blank>, it chose to fill the blank with a conjunction producing *the destruction of the city* **and**. The frequency of syntactic codes in the corpus following the syntactic codes for *the city*, *of the city*, *destruction of the city* or *the destruction of the city* were always greater (sometimes 10 times greater) for other syntactic categories (e.g. prepositions) than for conjunctions.

Although, we don't have a complete analysis of the constituents PARSNIP knows, we do have some evidence to suggest that PARSNIP recognizes noun-phrases and other higher order consituents in the hidden layer. PARSNIP was exposed to sentence fragments that were noun phrases, verb phrases, or random sentence fragments. Then, during recognition, the hidden layer values were clustered yielding groups containing either noun phrases with some random fragments or verb phrases. Much more can be done with this type of methodology in terms of isolating the constituent information that PARSNIP uses.

The acquisition strategies exhibited by PARSNIP conform to what is usually thought of as a nativist principle, the Subset Principle. This principle is usually described in terms of a child moving from one grammar to another:

> "Each step of a child's acquisition of grammar must involve movement from a smaller set to a larger set and cannot involve the reverse. The steps are motivated by pieces of input data (adult sentences) which fail to fit into the smaller set, thereby forcing an expansion of the set." (Roeper, in press)

This is exactly the sort of conservative generalization that one might expect from an auto-associator such as the one employed by PARSNIP. The network is lead to change its syntactic knowledge (connections/weights) based solely on single sentence violations of prior successful generalizations about a subset of sentences that it had previously constructed. This process is incremental because the entire learning process in connectionist networks is based on small incremental changes motivated the success or failure of its generalizations about the data.

References

Ackley, D. Hinton G.E. and Sejnowski, T., A Learning Algorithm for Boltzmann Machines, Cognitive Science, 9,1,1985.

Berwick, B. The acquisition of syntactic knowledge, MIT Press, 1985.

Chomsky, N. Syntactic structures, The Hague: Mouton, 1957.

Chomsky, N. Aspects of the Theory of Syntax, Cambridge, Mass.: MIT Press, 1965.

Chomsky, N. Language and Mind, Harcourt Brace Jovanovich, 1972.

Chomsky, N. Lectures on Government and Binding, Foris, 1981.

Feldman, J. Connectionist Models and Their Applications. Cognitive Science Special Issue, 9, 1, 1985.

Francis, W. N. and Kučera H., Manual of information to accompany a standard corpus of present-day edited american english for use with digital computers, Department of Linguistics, Brown University, 1979.

Hanson, S. J. and Burr, D. J., Knowledge Representation in Connectionist Networks. Submitted paper to AAAI, 1987.

Miller, G. A., Some psychological studies of grammar, American Psychologist, 17, 748-762, 1962.

McClelland J. and Kawamoto A. H., Mechanisms of sentence processing: assigning roles to constituents, in Parallel Distributed Processing, Vol II: Psychological and Biological Models, McClelland J. and Rumelhart D. (Eds.), Bradford Books/MIT Press, 1986.

Radford, A., Transformational Syntax, New York: Cambridge University Press, 1981.

Randall, J. H., Morphological Structure and Language Acquisition, Unpublished Doctoral Dissertation, University of Mass. at Amherst, Linguistics Department, 1982.

Roeper, T. Formal and substantive features of language acquisition: reflections on the subset principle and parametric variation, in Cognitive Science, S. Steele (Ed.) University of Arizona Press, in press.

Rumelhart D.E., Hinton G.E., and Williams R., Learning Internal Representations by error propagation. Nature, 1986.

Rumelhart D. E. and McClelland J. (Eds.), Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Vol 1: Foundations. Bradford Books/MIT Press, Cambridge, Mass., 1986

Rumelhart D. E. and McClelland J., On learning the past tenses of english verbs, in Parallel Distributed Processing, Vol II: Psychological and Biological Models, McClelland J. and Rumelhart D. (Eds.), Bradford Books/MIT Press, 1986.

Selman, B. Rule-based processing in a connectionist system for natural language understanding (TR CSRI-168), Toronto: University of Toronto, Computer Systems Research Institute, 1985.