Finite models of infinite language:
A connectionist approach to recursion

Morten H. Christiansen
Southern Illinois University, Carbondale

Nick Chater
University of Warwick

Running head: Finite models of language

Address for correspondence:
Morten H. Christiansen
Department of Psychology
Southern Illinois University
Carbondale, IL 62901-6502
Phone: (618) 453-3547
Fax: (618) 453-3563
Email: *morten@siu.edu*

1 Introduction

In linguistics and psycholinguistics, it is standard to assume that natural language involves rare but important recursive constructions. This assumption originates with Chomsky's (1957, 1959, 1965) arguments that the grammars for natural languages exhibit potentially unlimited recursion. Chomsky assumed that, if the grammar allows a recursive construction, it can apply arbitrarily many times. Thus, if (1) is sanctioned with one level of recursion, then the grammar must sanction arbitrarily many levels of recursion, generating, for example, (2) and (3).

(1) *The mouse that the cat bit ran away.*

(2) *The mouse that the cat that the dog chased bit ran away.*

(3) *The mouse that the cat that the dog that the man frightened chased bit ran away.*

But people can only deal easily with relatively simple recursive structures (e.g., Bach, Brown & Marslen-Wilson, 1986). Sentences like (2) and (3) are extremely difficult to process.

Note that the idea that natural language is recursive requires broadening the notion of which sentences are in the language, to include sentences like (2) and (3). To resolve the difference between language so construed and the language that people produce and comprehend, Chomsky (e.g., 1965) distinguished between linguistic *competence* and human *performance*. Competence refers to a speaker/hearer's knowledge of the language, as studied by linguistics. In contrast, psycholinguists study performance—i.e., how linguistic knowledge is used in language processing, and how non-linguistic factors interfere with using that knowledge. Such "performance factors" are invoked to explain why some sentences, while consistent with linguistic competence, will not be said or understood.

The claim that language allows unbounded recursion has two key implications. First, processing unbounded recursive structures requires unlimited memory—this rules out finite state models of language processing. Second, unbounded recursion was said to require innate knowledge because the child's language input contain so few recursive constructions. These implications struck at the heart of the then-dominant approaches to language. Both structural linguistics and behaviorist psychology (e.g., Skinner 1957) lacked the generative mechanisms to explain unbounded recursive structures. And the problem of learning recursion undermined both the learning mechanisms described by the behaviorists, and the corpus-based methodology of structural linguistics. More importantly, for current cognitive science, both problems appear to apply to connectionist models of language. Connectionist networks consist of finite sets of processing units, and therefore appear to constitute a finite state model of language, just as behaviorism assumed; and connectionist models learn by a kind of associative learning algorithm, more elaborate than, but similar in spirit to, that postulated by behaviorism. Furthermore, connectionist models attempt to learn the structure of the language from finite corpora, echoing the corpus-based methodology of structural linguistics. Thus, it seems that Chomsky's arguments from the 1950s and 1960s may rule out, or at least, limit the scope of, current connectionist models of language processing.

One defense of finite state models of language processing, to which the connectionist might turn, is that connectionist models should be performance models, capturing the limited recursion people can process, rather than the unbounded recursion of linguistic competence (e.g., Christiansen, 1992), as the above examples illustrate. Perhaps, then, finite state models can model actual human language processing successfully.

This defense elicits a more sophisticated form of the original argument: that what is important about generative grammar is not that it allows arbitrarily complex strings, but that it gives simple rules capturing *regularities* in language. An adequate model of language processing must somehow embody grammatical knowledge that can capture these regularities. In symbolic computational linguistics, this is done by representing grammatical information and processing operations as symbolic rules. While these rules could, in principle, apply to sentences of arbitrary length and complexity, in practice they are bounded by the finiteness of the underlying hardware. Thus, a symbolic model of language processing, such as CC-READER (Just & Carpenter, 1992), embodies the competence-performance distinction in this way. Its grammatical competence consists of a set of recursive production rules which are applied to produce state changes in a working memory. Limitations on the working memory's capacity explain performance limitations without making changes to the competence part of the model. Thus a finite processor, CC-READER, captures underlying recursive structures. Unless connectionist networks can perform the same trick they cannot be complete models of natural language processing.

From the perspective of cognitive modeling, therefore, the unbounded recursive structure of natural language is not axiomatic. Nor need the suggestion that a speaker/hearer's knowledge of the language captures such infinite recursive structure be taken for granted. Rather, the view that "unspeakable" sentences which accord with recursive rules form a part of the knowledge of language is an *assumption* of the standard view of language pioneered by Chomsky and now dominant in linguistics and much of psycholinguistics. The challenge for a connectionist model is to account for those aspects of human comprehension/production performance that suggest the standard recursive picture. If connectionist models can do this without making the assumption that the language processor really implements recursion, or that arbitrarily complex recursive structures really are sentences of the language, then they may present a viable, and radical, alternative to the standard 'generative' view of language and language processing.

Therefore, in assessing the connectionist simulations that we report below, which focuses on natural language recursion, we need not require that connectionist systems be able to handle recursion in full generality. Instead, the benchmark for performance of connectionist systems will be set by human abilities to handle recursive structures. Specifically, the challenge for connectionist researchers is to capture the recursive regularities of natural language, while allowing that arbitrarily complex sentences cannot be handled. This requires (a) handling recursion at a comparable level to human performance, and (b) learning from exposure and generalizing to novel recursive constructions. Meeting this challenge involves providing a new account of people's limited ability to handle natural language recursion, without assuming an internally represented grammar which allows unbounded recursion—i.e., without invoking the competence/performance distinction.[1]

Here, we consider natural language recursion in a highly simplified form. We train connectionist networks on small artificial languages that exhibit the different types of recursion in natural language. This addresses directly Chomsky's (1957) arguments that recursion in natural language *in principle* rules out associative and finite state models of language processing. Considering recursion in a pure form permits us to address the in-principle viability of connectionist networks in handling recursion, just as simple artificial languages have been used to assess the feasibility of symbolic parameter-setting approaches to language acquisition (Gibson & Wexler, 1994; Niyogi & Berwick, 1996).

The structure of this chapter is as follows. We begin by distinguishing varieties of recursion in natural language. We then summarize past connectionist research on natural language recursion. Next, we introduce three artificial languages, based on Chomsky's (1957) three kinds of recursion, and describe the performance of connectionist networks trained on these languages. These results suggest that the networks handle recursion to a degree comparable with humans. We close with conclusions for the prospects of connectionist models of language processing.

2 Varieties of Recursion

Chomsky (1957) introduced the notion of a recursive generative grammar. Early generative grammars were assumed to consisted of phrase structure rules and transformational rules (which we shall not consider below). Phrase structure rules have the form $A \rightarrow BC$, meaning that the symbol A can be replaced by the concatenation of B and C. A phrase structure rule is *recursive* if a symbol X is replaced by a string of symbols which includes X itself (e.g., $A \rightarrow BA$). Recursion can also arise through applying recursive *sets* of rules, none of which need individually be recursive. When such rules are used successively to expand a particular symbol, the original symbol may eventually be derived. A language construction modeled using recursion rules is a recursive *construction*; a *language* has recursive structure if it contains such constructions.

Modern generative grammar employs many formalisms, some distantly related to phrase structure rules. Nevertheless, corresponding notions of recursion within those formalisms can be defined. We shall not consider such complexities here, but use phrase structure grammar throughout.

There are several kinds of recursion relevant to natural language. First, there are those generating languages that could equally well be generated non-recursively, by iteration. For example, the rules for *right-branching* recursion shown in Table 1 can generate the right-branching sentences (4)–(6):

(4) *John loves Mary.*

(5) *John loves Mary who likes Jim.*

(6) *John loves Mary who likes Jim who dislikes Martha.*

But these structures can be produced or recognized by a finite state machine using iteration. The recursive structures of interest to Chomsky, and of interest here, are those where recursion is indispensable.

——————insert Table 1 about here——————

Chomsky (1957) invented three artificial languages, generated by recursive rules from a vocabulary consisting only of *a*'s and *b*'s. These languages cannot be generated or parsed by a finite state machine. The first language, which we call *counting recursion*, was inspired by sentence constructions like *'if S_1, then S_2'* and *'either S_1, or S_2'.* These can, Chomsky assumed, be nested arbitrarily, as in (7)–(9):

(7) *if S_1 then S_2.*

(8) *if if S_1 then S_2 then S_3.*

(9) *if if if S_1 then S_2 then S_3 then S_4.*

The corresponding artificial language has the form $a^n b^n$ and includes the following strings:

(10) *ab, aabb, aaabbb, aaaabbbb, aaaaabbbbb, . . .*

Unbounded counting recursion cannot be parsed by any finite device processing from left to right, because the number of '*a*'s must be stored, and this can be unboundedly large, and hence can exceed the memory capacity of any finite machine.

The second artificial language was modeled on the center-embedded constructions in many natural languages. For example, in sentences (1)–(3) above the dependencies between the subject nouns and their respective verbs are center-embedded, so that the first noun is matched with the last verb, the second noun with the second but last verb, and so on. The artificial language captures these dependency relations by containing sentences that consists of a string X of *a*'s and *b*'s followed by a 'mirror image' of X (with the words in the reverse order), as illustrated by (11):

(11) *aa, bb, abba, baab, aaaa, bbbb, aabbaa, abbbba, . . .*

Chomsky (1957) used the existence of center-embedding to argue that natural language must be at least context-free, and beyond the scope of any finite machine.

The final artificial language resembles a less common pattern in natural language, *cross-dependency*, which is found in Swiss-German and in Dutch,[2] as in (12)-(14) (from Bach, Brown & Marslen-Wilson, 1986):

(12) *De lerares heeft de knikkers opgeruimd.*

Literal: The teacher has the marbles collected up

Gloss: *The teacher collected up the marbles.*

(13) *Jantje heeft de lerares de knikkers helpen opruimen.*

Literal: Jantje has the teacher the marbles help collect up.

Gloss: *Jantje helped the teacher collect up the marbles.*

(14) *Aad heeft Jantje de lerares de knikkers laten helpen opruimen.*

Literal: Aad has Jantje the teacher the marbles let help collect up.

Gloss: *Aad let Jantje help the teacher collect up the marbles.*

Here, the dependencies between nouns and verbs are crossed such that the first noun matches the first verb, the second noun matches the second verb, and so on. This is captured in the artificial language by having all sentences consist of a string X followed by an identical copy of X as in (15):

(15) *aa, bb, abab, baba, aaaa, bbbb, aabaab, abbabb, . . .*

The fact that cross-dependencies cannot be handled using a context-free phrase structure grammar has meant that this kind of construction, although rare even in languages in which it occurs, has assumed considerable importance in linguistics.[3] Whatever the linguistic status of complex recursive constructions, they are difficult to process compared to right-branching structures. Structures analogous to counting recursion have not been studied in psycholinguistics, but sentences such as (16), with just one level of recursion, are plainly difficult (see Reich, 1969).

(16) *If if the cat is in, then the dog cannot come in then the cat and dog dislike each other.*

The processing of center-embeddings has been studied extensively, showing that English sentences with more than one center-embedding (e.g., sentences (2) and (3) presented above) are read with the same intonation as a list of random words (Miller, 1962), that they are hard to memorize (Foss & Cairns, 1970; Miller & Isard, 1964), and that they are judged to be ungrammatical (Marks, 1968). Using sentences with semantic bias or giving people training can improve performance on such structures, to a limited extent (Blaubergs & Braine, 1974; Stolz, 1967). Cross-dependencies have received less empirical attention, but present similar processing difficulties to center-embeddings (Bach et al., 1986; Dickey & Vonk, 1997).

3 Connectionism and Recursion

Connectionist models of recursive processing fall in three broad classes. Some early models of syntax dealt with recursion by "hardwiring" symbolic structures directly into the network (e.g., Fanty, 1986; Small, Cottrell & Shastri, 1982). Another class of models attempted to learn a grammar from "tagged" input sentences (e.g., Chalmers, 1990; Hanson & Kegl, 1987; Niklasson & van Gelder, 1994; Pollack, 1988, 1990; Stolcke, 1991). Here, we concentrate on a third class of models that attempts the much harder task of learning syntactic structure from strings of words (see Christiansen & Chater, Chapter 2, this volume, for further discussion of connectionist sentence processing models). Much of this work has been carried out using the Simple Recurrent Network (SRN) (Elman, 1990) architecture. The SRN involves a crucial modification to a standard feedforward network–a so-called "context layer"—allowing past internal states to influence subsequent states (see Figure 1 below). This provides the SRN with a memory for past input, and therefore an ability to process input sequences, such as those generated by finite-state grammars (e.g., Cleeremans, Servan-Schreiber & McClelland, 1989; Giles, Miller, Chen, Chen, Sun & Lee, 1992; Giles & Omlin, 1993; Servan-Schreiber, Cleeremans & McClelland, 1991).

Previous efforts in modeling complex recursion fall in two categories: simulations using language-like grammar fragments and simulations relating to formal language theory. In the first category, networks are trained on relatively simple artificial languages, patterned on English. For example, Elman (1991, 1993) trained SRNs on sentences generated by a small context-free grammar incorporating center-embedding and one kind of right-branching recursion. Within the same framework, Christiansen (1994, 2000) trained SRNs on a recursive artificial language incorporating four kinds of right-branching structures, a left branching structure, and center-embedding or cross-dependency. Both found that network performance degradation on complex recursive structures mimicked human behavior (see Christiansen & Chater, Chapter 2, this volume, for further discussion of SRNs as models of language processing). These results suggest that SRNs can capture the quasi-recursive structure of actual spoken language. One of the contributions of the present chapter is to show that the SRN's general pattern of performance is relatively invariant over variations in network parameters and training corpus—thus, we claim, the human-like pattern of performance arises from *intrinsic* constraints of the SRN architecture.

While work in the first category has been suggestive but relatively unsystematic, work in the second category has involved detailed investigations of small artificial tasks, typically using very small networks. For example, Wiles and Elman (1995) made a detailed study of counting recursion, with a recurrent networks with 2 hidden units (HU),[4] and found a network that generalized to inputs far longer than those used in training. Batali (1994) used the same language, but employed 10HU SRNs and showed that networks could reach good levels of performance, when selected by a process of "simulated evolution" and then trained using conventional methods. Based on a mathematical analysis, Steijvers and Grünwald (1996) "hardwired" a second order 2HU recurrent network (Giles et al., 1992) to process the context-sensitive counting language $b(a)^k b(a)^k \ldots$ for values of k between 1 and 120. An interesting question, which we address below, is whether performance changes with more than two vocabulary items—e.g., if the network must learn to assign items into different lexical categories ("noun" and "verb") as well as paying attention to dependencies between these categories. This question is important with respect to the relevance of these results for natural language processing.

No detailed studies have previously been conducted with center-embedding or crossed-dependency constructions. The studies below comprehensively compare all three types of recursion discussed in Chomsky (1957), with simple right-branching recursion as a baseline. Using these abstract languages allows recursion to be studied in a "pure" form, without interference from other factors. Despite the idealized nature of these languages, the SRN's performance qualitatively conforms to human performance on similar natural language structures.

A novel aspect of these studies is comparison with performance benchmark from statistical linguistics. The benchmark method is based on n-grams, i.e., strings of n consecutive words. It is "trained" on the same input as the networks, and records the frequency of each n-gram. It predicts new words from the relative frequencies of the n-grams which are consistent with the previous $n-1$ words. The prediction is a vector of relative frequencies for each possible successor item, scaled to sum to 1, so that they can be interpreted as probabilities, and are comparable with the output vectors of the networks. Below, we compare network performance with the predictions of bigram and trigram models.[5] These simple models can provide insight into the sequential information which the networks pick up, and make a link with statistical linguistics (e.g., Charniak, 1993).

4 Three Benchmark Tests Concerning Recursion

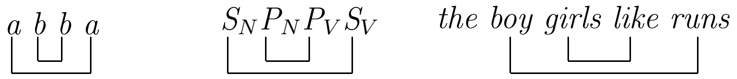
We constructed three languages to provide input to the network. Each language has two recursive structures: one of the three complex recursive constructions and the right-branching construction as a baseline. Vocabulary items were divided into "nouns" and "verbs", incorporating both singular or plural forms. An end of sentence marker (EOS) completes each sentence.

i. *Counting recursion*

aabb *NNVV*

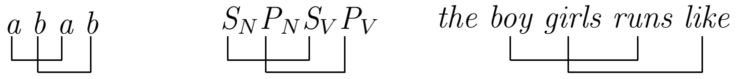
For counting recursion, we treat Chomsky's symbols *'a'* and *'b'* as the categories of noun and verb, respectively, and ignore singular/plural agreement.

ii. *Center-embedding recursion*



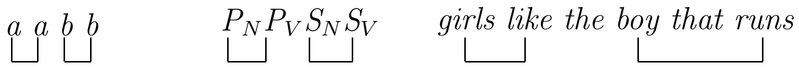
In center-embedding recursion, we map *'a'* and *'b'* onto the categories of singular and plural words (whether nouns or verbs). Nouns and verbs agree for number as in center-embedded constructions in natural language.

iii. *Cross-dependency recursion*



In cross-dependency recursion we map *'a'* and *'b'* onto the categories of singular and plural words. Nouns and verbs agree for number as in cross-dependency constructions.

iv. *Right-branching recursion*



For right-branching recursion, we map *'a'* and *'b'* onto the categories of singular and plural words. Nouns and verbs agree as in right-branching constructions.

Thus, the counting recursive language consisted of both counting recursive constructions (i) interleaved with right-branching recursive constructions (iv), the center-embedding recursive language of center-embedded recursive constructions (ii) interleaved with right-branching recursive constructions (iv), and the cross-dependency recursive language of cross-dependency recursive constructions (iii) interleaved with right-branching recursive constructions (iv).

How can we assess how well a network has learned these languages? By analogy with standard linguistic methodology, we could train the net to make "grammaticality judgments", i.e., to distinguish legal and non-legal sentences. But this chapter focuses on *performance* on recursive structures, rather than meta-linguistic judgments (which are often assumed to relate to linguistic competence).[6] Therefore, we use a task which directly addressed how the network processes sentences, rather than requiring it to make meta-linguistic judgments. Elman (1990) suggested such an approach, which has become standard in SRN studies of natural language processing. The network is trained to predict the *next* item in a sequence, given previous context. That is, the SRN gets an input word at time t and then predicts the word at $t + 1$. In most contexts in real natural language, as in these simulations, prediction will not be perfect. But while it is not possible to be certain what item will come next, it is possible to predict successfully which items are *possible* continuations, and which are not, according to the regularities in the corpus. To the extent that the network can predict successfully, then, it is learning the regularities underlying the language.

5 Simulation Results

We trained SRNs on the three languages, using a sixteen word vocabulary with four singular nouns, four singular verbs, four plural nouns, and four plural verbs.[7] All nets had 17 input and output units (see Figure 1), where units correspond to words, or the EOS marker. The hidden layer contained between 2 and 100 units. Except where noted below, training corpora consisted of 5000 variable length sentences, and test corpora of 500 novel sentences, generated in the same way. The training and test corpora did not overlap. Each corpus was concatenated into a single long string and presented to the network word by word. Both training and test corpora comprised 50% complex recursive constructions interleaved with 50% right-branching constructions. The distribution of depth of embedding is shown in Table 2. The mean sentence length in training and test corpora was 4.7 words (SD: 1.3).

————insert Figure 1 about here————

————insert Table 2 about here————

Since the input consists of a single concatenated string of words, the network has to discover that the input consists of sentences, i.e., nouns followed by verbs (ordered by the constraints of the language being learned) and delineated by EOS markers. Consider an SRN trained on the center-embedding language and presented with the two sentences: '$n_1 v_5 \# N_3 n_8 v_2 V_4 \#$'.[8] First, the network gets 'n_1' as input and is expected to produce 'v_5' as output. The weights are then adjusted depending on the discrepancy between the actual and desired output and the desired output using back-propagation (Rumelhart, Hinton & Williams, 1986). Next, the SRN receives 'v_5' as input and should produce as output the end-of-sentence marker ('#'). At the next time-step, '#' is provided as input and 'N_3' is the target output, followed by the input/output pairs: 'N_3/n_8', 'n_8'/'v_2', 'v_2'/'V_4', and 'V_4'/'#'. Training continues in this manner for the whole training corpus.

Test corpora were then presented to the SRNs and output recorded, with learning turned off. As we noted above, in any interesting language-like task, the next item is not deterministically specified by the previous items. In the above example at the start of the second sentence, the grammar for the center-embedding language permits both noun categories, 'n' and 'N', to begin a sentence. If the SRN has acquired the relevant aspects of the grammar which generated the training sentences, then it should activate all word tokens in both 'n' and 'N' following an EOS marker. Specifically, the network's optimal output is the conditional probability distribution over possible next items. We can therefore measure amount of learning by comparing the network's output with an estimate of the true conditional probabilities (this gives a less noisy measure than comparing against actual next items). This overall performance measure is used next. Below, we introduce a measure of Grammatical Prediction Error to evaluate performance in more detail.

5.1 Overall performance

As we noted, our overall performance measure compared network outputs with estimates of the true conditional probabilities given prior context, which, following Elman (1991), can be estimated from the training corpus. However, such estimates cannot assess performance on novel test sentences, because a naive empirical estimate of the probability of any novel sentence is 0, as it has never previously occurred. One solution to this problem is to estimate the conditional probabilities based on the prior occurrence of lexical categories—e.g., 'NVnvnvNV#'—rather than individual words. Thus, with $\mathbf{c_i}$ denoting the category of the ith word in the sentence we have the following relation:[9]

————insert equation 1 here————

where the probability of getting some member of a given lexical category as the pth item, $\mathbf{c_p}$, in a sentence is conditional on the previous $p-1$ lexical categories. Note that for the purpose of performance assessment singular and plural nouns are assigned to separate lexical categories throughout this chapter, as are singular and plural verbs.

Given that the choices of lexical item for each category are independent, and that each word in the category is equally frequent,[10] the probability of encountering a word $\mathbf{w_n}$, which is a member of a category $\mathbf{c_p}$, is inversely proportional to the number of items, $\mathbf{C_p}$, in that category. So, overall,

————insert equation 2 here————

If the network is performing optimally, the output vector should exactly match these probabilities. We measure network performance by the summed squared difference between the network outputs and the conditional probabilities, defining Squared Error:

——————insert equation 3 here——————

where W is the set of words in the language (including the end of sentence marker), and there is an output unit of the network corresponding to each word. The index j runs through each possible next word, and compares the network output to the conditional probability of that word. Finally, we obtain an overall measure of network performance by calculating the Mean Squared Error (MSE) across the whole test corpus. MSE will be used as a global measure of the performance of both networks and n-gram models below.

5.1.1 Intrinsic constraints on SRN performance

Earlier simulations concerning the three languages (Christiansen, 1994) showed that performance degrades as embedding depth increases. As mentioned earlier, SRN simulations in which center-embeddings were included in small grammar fragments have the same outcome (Christiansen, 1994, 2000; Elman, 1991, 1993; Weckerly & Elman, 1992) and this is also true for cross-dependencies (Christiansen, 1994, 2000). But does this human-like pattern arise intrinsically from the SRN architecture, or is it an artifact of the number of HUs used in typical simulations?

To address this objection, SRNs with 2, 5, 10, 15, 25, 50, and 100 HUs were trained on the three artificial languages. Across all simulations, the learning rate was 0.1, no momentum was used, and the initial weights were randomized to values in the interval [-0.25,0.25]. Although the results presented here were replicated across different initial weight randomizations, we focus on a typical set of simulations for the ease of exposition. Networks of the same size were given the same initial random weights to facilitate comparisons across the three languages.

Figure 2 shows performance averaged across epochs for different sized nets tested on corpora consisting entirely of either complex recursive structures (left panels) or right-branching recursive structures (right panels). All test sentences were novel and varied in length (following the distribution in Table 2). The MSE values were calculated as the average of the MSEs sampled at every second epoch (from epoch 0 to epoch 100). The MSE for bigram and trigram models are included (black bars) for comparison.

——————insert figure 2 about here——————

The SRNs performed well. On counting recursion, nets with 15HUs or more obtained low MSE on complex recursive structures (top left panel). Performance on right-branching structures (top right panel) was similar across different numbers of HUs. For both types of recursion, the nets outperformed the bigram and trigram models. For the center-embedding language, nets with at least 10HUs achieved essentially the same level of performance on complex recursive structures (middle left panel), whereas nets with five or more HUs performed similarly on the right-branching structures (middle right panel). Again, the SRNs generally outperformed bigram and trigram models. Nets with 15HUs or more trained on the cross-dependency language all reached the same level of performance on complex recursive structures (bottom left panel). As with counting recursion, performance was quite uniform on right-branching recursive constructions (bottom right panel) for all numbers of HUs, and the SRNs again outperformed bigram and trigram models. These results suggest that the above objection does not apply to the SRN. Above 10-15HUs, the number of HUs seems not to affect performance.

Comparing across the three languages, the SRN found the counting recursion language the easiest and found cross-dependencies easier than center embeddings. This is important because people also appear better at dealing with cross-dependency constructions than equivalent center-embedding constructions. This is surprising for linguistic theory where cross-dependencies are typically viewed as more complex than center-embeddings because, as we noted above, they cannot be captured by phrase-structure rules. Interestingly, the bigram and trigram models showed the opposite effect, with better performance on center-embeddings than cross-dependencies. Finally, the SRNs with at least 10 hidden units had a lower MSE on complex recursive structures than on right-branching structures. This could be because the complex recursive constructions essentially become deterministic (with respect to length) once the first verb is encountered, but this is not generally true for right-branching constructions.

These results show that the number of HUs, when sufficiently large, does not substantially influence performance on these test corpora. Yet perhaps the number of HUs may matter when processing the doubly embedded complex recursive structures which are beyond the limits of human performance. To assess this, Christiansen and Chater (1999) retested the SRNs (trained on complex and right-branching constructions of varying length) on corpora containing just novel doubly embedded structures. Their results showed a similar performance uniformity to that in Figure 2. These simulations also demonstrated that once an SRN has a sufficient size (5-10 HUs) it outperforms both n-gram models on doubly embedded constructions. Thus, above a sufficient number of hidden units, the size of the hidden layer does seems irrelevant to performance on novel doubly embedded complex constructions drawn from the three languages. Two further objections may be raised, however.

First, perhaps the limitations on processing complex recursion is due to the interleaving of right-branching structures during training. To investigate this objection, SRNs with 2, 5, 10, 15, 25, 50 and 100 HUs were trained (with the same learning parameters as above) on versions of the three languages only containing complex recursive constructions of varying length. When tested on complex recursive sentence structures of varying length, the results were almost identical to those in the left panels of Figure 2, with a very similar performance uniformity across the different HU sizes (above 5-10 units) for all three languages. Also as before, this performance uniformity was also evident when on corpora consisting entirely of doubly embedded complex constructions. Moreover, similar results were found for SRNs of different HU sizes trained on a smaller 5 word vocabulary. These additional simulations show that the interleaving of the right-branching constructions does not significantly alter performance on complex recursive constructions.

Second, perhaps processing limitations result from an inefficient learning algorithm. An alternative training regime for recurrent networks, back-propagation through time (BPTT), appears preferable on theoretical grounds, and is superior to SRN training in various artificial tasks (see Chater & Conkey, 1992). But choice of learning algorithm does not appear to be crucial here. Christiansen (1994) compared the SRN and BPTT learning algorithms on versions of the three languages only containing complex recursive constructions of varying length (and same embedding depth distribution as in Table 2). In one series of simulations, SRNs and BPTT training (unfolded 7 steps back in time) with 5, 10 and 25 HUs were trained using a five word vocabulary. There was no difference across the three languages between SRN and BPTT training. Further simulations replicated these results for nets with 20HUs and a 17 word vocabulary. Thus, there is currently no evidence that the human level processing limitations that are exhibited in these simulations are artifacts of using an inefficient learning algorithm.

5.2 Performance at different depths of embedding

We have seen that the overall SRN performance roughly matches human performance on recursive structures. We now consider performance at different levels of embedding. Human data suggests that performance should degrade rapidly as embedding depth increases for complex recursive structures, but that it should degrade only slightly for right-branching constructions.

Above we used empirical conditional probabilities based on lexical categories to assess SRN performance (Equations 2 and 3). However, this measure is not useful for assessing performance on novel constructions which either go beyond the depth of embedding found in the training corpus, or deviate, as ungrammatical forms do, from the grammatical structures encountered during training. For comparisons with human performance we therefore use a different measure: Grammatical Prediction Error (GPE).

When evaluating how the SRN has learned the grammar underlying the training corpus, it is not only important to determine whether the words the net predicts are grammatical, but also that the net predicts all the possible grammatical continuations. GPE indicates how a network is obeying the training grammar in making its predictions, taking hits, false alarms, correct rejections and misses into account. Hits and false alarms are calculated as the accumulated activations of the set of units, G, that are grammatical and the set of ungrammatical activated units, U, respectively:

————insert equation 4 here————

————insert equation 5 here————

Traditional sensitivity measures, such as *d'* (Signal Detection Theory, Green & Swets, 1966) or α (Choice Theory, Luce, 1959), assume that misses can be calculated as the difference between total number of relevant observations and hits. But, in terms of network activation, "total number of relevant observations" has no clear interpretation.[11] Consequently, we need an alternative means of quantifying misses; that is, to determine an activation-based penalty for not activating all grammatical units and/or not allocating sufficient activation to these units. With respect to GPE, the calculation of misses involves the notion of a target activation, t_i, computed as a proportion of the total activation (hits and false alarms) determined by the lexical frequency, f_i, of the word that unit i designates and weighted by the sum of the lexical frequencies, f_j, of all the grammatical units:

————insert equation 6 here————

The missing activation for each unit can be determined as the positive discrepancy, m_i, between the target activation, t_i, and actual activation, u_i, for a grammatical unit:

——————insert equation 7 here——————

Finally, the total activation for misses is the sum over missing activation values:

——————insert equation 8 here——————

The GPE for predicting a particular word given previous sentential context is thus measured by:

——————insert equation 9 here——————

GPE measures how much of the activation for a given item accords with the grammar (hits) in proportion to the total amount of activation (hits and false alarms) and the penalty for not activating grammatical items sufficiently (misses). Although not a term in Equation 9, correct rejections are taken into account by assuming that they correspond to zero activation for units that are ungrammatical given previous context.

GPEs range from 0 and 1, providing a stringent measure of performance. To obtain a perfect GPE of 0 the SRN must predict all and only the next items prescribed by the grammar, scaled by the lexical frequencies of the legal items. Notice that to obtain a low GPE the network must make the correct subject noun/verb agreement predictions (Christiansen & Chater, 1999). The GPE value for an individual word reflects the difficulty that the SRN experienced for that word, given the previous sentential context. Previous studies (Christiansen, 2000; MacDonald & Christiansen, in press) have found that individual word GPE for an SRN can be mapped qualitatively onto experimental data on word reading times, with low GPE reflecting short reading times. Average GPE across a sentence measures the difficulty that the SRN experienced across the sentence as a whole. This measure maps onto sentence grammaticality ratings, with low average GPEs indicating high rated "goodness" (Christiansen & MacDonald, 2000). .

5.2.1 Embedding depth performance

We now use GPE to measure SRN performance on different depths of embedding. Given that number of HUs seems relatively unimportant, we focus just on 15HU nets below. Inspection of MSE values across epochs revealed that performance on complex recursive constructions asymptotes after 35–40 training epochs. From the MSEs recorded for epochs 2 through 100, we chose the number of epochs at which the 15HU nets had the lowest MSE. The best level of performance was found after 54 epochs for counting recursion, 66 epochs for center-embedding, and 92 epochs for cross-dependency. Results reported below use SRNs trained for these number of epochs.

Figure 3 plots average GPE on complex and right-branching recursive structures against embedding depth for 15HU nets, bigram models, and trigram models (trained on complex and right-branching constructions of varying length). Each data point represents the mean GPE on 10 novel sentences. For the SRN trained on counting recursion there was little difference between performance on complex and right-branching recursive constructions, and performance only deteriorated slightly with increasing embedding depth. In contrast, the n-gram models (and especially the trigram model) performed better on right-branching structures than complex recursive structures. Both n-gram models showed a sharper decrease in performance across depth of recursion than the SRN. The SRN trained on center-embeddings also outperformed the n-gram models, although it, too, had greater difficulty with complex recursion than with right-branching structures. Interestingly, SRN performance on right-branching recursive structures decreased slightly with depth of recursion. This contrasts with many symbolic models where unlimited right-branching recursion poses no processing problems (e.g., Church, 1982; Gibson, 1998; Marcus, 1980; Stabler, 1994). However, the performance deterioration of the SRN appears in line with human data (see below). A comparison between the n-gram models' performance on center-embedding shows that whereas both exhibited a similar pattern of deteriorating with increasing depth on the complex recursive constructions, the trigram models performed considerably better on the right-branching constructions than the bigram model. As with the MSE results presented above, SRN performance on cross-dependencies was better than on center-embeddings. Although the SRN, as before, obtained lower GPEs on right-branching constructions compared with complex recursive structures, the increase in GPE across embedding depth on the latter was considerably less for the cross-dependency net than for its center-embedding counterpart. Bigrams performed poorly on the cross-dependency language both on right-branching and complex recursion. Trigrams performed substantially better, slightly outperforming the SRN on right branching structures, though still lagging behind the SRN on complex recursion. Finally, note that recursive depth 4 was not seen in training. Yet there was no abrupt breakdown in performance for any of the three languages at this point, for both SRNs and n-gram models. This suggests that these models are able to generalize to at least one extra level of recursion beyond what they have been exposed to during training (and this despite only 1% of the training items being of depth 3).

——————insert figure 3 about here——————

Overall, the differential SRN performance on complex recursion and right-branching constructions for center-embeddings and cross-dependencies fit well with human data.[12]

5.3 Training exclusively on doubly embedded complex constructions

An alternative objection to the idea of intrinsic constraints being the source of SRN limitations is that these limitations might stem from the statistics of the training corpora: e.g., perhaps the fact that just 7% of sentences involved doubly embedded complex recursive structures explains the poor SRN performance with these structures. Perhaps adding more doubly embedded constructions would allow the SRN to process these constructions without difficulty.

We therefore trained 15HU SRNs on versions of the three languages consisting exclusively of doubly embedded complex recursion without interleaving right-branching constructions. Using the same number of words as before, best performance was found for the counting recursion depth 2 trained SRN (D2-SRN) after 48 epochs, after 60 epochs for the center-embedding D2-SRN, and after 98 epochs for the cross-dependency D2-SRN. When tested on the test corpora containing only novel doubly embedded sentences, the average MSE found for the counting recursion network was 0.045 (vs. 0.080 for the previous 15HU SRN), 0.066 for the center-embedding net (vs. 0.092 for the previous 15HU SRN), and 0.073 for the cross-dependency net (vs. 0.079 for the previous 15HU SRN). Interestingly, although there were significant differences between the MSE scores for the SRNs and D2-SRNs trained on the counting recursion ($t(98) = 3.13, p < 0.003$) and center-embeddings ($t(98) = 3.04, p < 0.004$), the difference between the two nets was not significant for cross-dependencies ($t(98) = .97, p > 0.3$). The performance of the D2-SRNs thus appear to be somewhat better than the performance of the SRNs trained on the corpora of varying length—at least for the counting and center-embedding recursion languages. However, D2-SRNs are only slightly better than their counterparts trained on sentences of varying length.

Figure 4 plots GPE against word position across doubly embedded complex recursive constructions from the three languages, averaged over 10 novel sentences. On counting recursion sentences (top panel), both SRN and D2-SRN performed well, with a slight advantage for the D2-SRN on the last verb. Both networks obtained lower levels of GPE than the bigrams and trigrams which were relatively inaccurate, especially for the last two verbs. On center-embeddings (middle panel), the two SRNs showed a gradual pattern of performance degradation across the sentence, but with the D2-SRN achieving somewhat better performance, especially on the last verb. Bigrams and trigrams performed similarly, and again performed poorly on the two final verbs. When processing doubly embedded cross-dependency sentences (bottom panel) SRN performance resembled that found for counting recursion. The GPE for both SRNs increased gradually, and close to each other, until the first verb. Then, the SRN GPE for the second verb dropped whereas the D2-SRN GPE continued to grow. At the third verb, the GPE for the D2-SRN dropped whereas the SRN GPE increased.

Although this pattern of SRN GPEs may seem puzzling, it appears to fit recent results concerning the processing of similar cross-dependency constructions in Dutch. Using a phrase-by-phrase self-paced reading task with stimuli adapted from Bach et al. (1986), Dickey and Vonk (1997) found a significant jump in reading times between the second and third verb, preceded by a (non-significant) decrease in reading times between the first and second verb. When the GPEs for individual words are mapped onto reading times, the GPE pattern of the SRN, but not the D2-SRN, provides a reasonable approximation of the pattern of reading times found by Dickey and Vonk. Returning to Figure 4, the trigram model—although not performing as well as the SRN—displayed a similar general pattern, whereas the bigram model performed very poorly. Overall, Figure 4 reveals that despite being trained exclusively on doubly embedded complex recursive constructions and despite not having to acquire the regularities underlying the right-branching structures, the D2-SRN only performed slightly better on doubly embedded complex recursive constructions than the SRN trained on both complex and right-branching recursive constructions of varying length. This suggests that SRN performance does not merely reflect the statistics of the training corpus, but intrinsic architectural constraints.

————insert figure 4 about here————

It is also interesting to note that the SRNs are not merely learning sub-sequences of the training corpus by rote—they substantially outperformed the *n*-gram models. This is particularly important because the material that we have used in these studies is the most favorable possible for n-gram models, since there is no intervening material at a given level of recursion. In natural language, of course, there is generally a considerable amount of material between changes of depth of recursion, which causes problems for n-gram models because they concentrate on short-range dependencies. While n-gram models do not generalize well to more linguistically natural examples of recursion, SRN models, by contrast, do show good performance on such material. We have found (Christiansen, 1994, 2000; Christiansen & Chater, 1994) that the addition of intervening non-recursive linguistic structure does not significantly alter the pattern of results found with the artificial languages reported here. Thus, SRNs are not merely learning bigrams and trigrams, but acquiring richer grammatical regularities that allow them to exhibit behaviors qualitatively similar to humans. We now consider the match with human data in more detail.

5.4 Fitting Human Data

5.4.1 Center-embedding vs. cross-dependency

As we have noted, Bach et al. (1986) found that cross-dependencies in Dutch were comparatively easier to process than center-embeddings in German. They had native Dutch speakers listen to sentences in Dutch involving varying depths of recursion in the form of cross-dependency constructions and corresponding right-branching paraphrases with the same meaning. Native German speakers were tested using similar materials in German, but with the cross-dependency constructions replaced by center-embedded constructions. Because of differing intuitions among German informants concerning whether the final verb should be in an infinitive or a past participle, two versions of the German materials were used. After each sentence, subjects rated its comprehensibility on a 9-point scale (1 = easy, 9 = difficult). Subjects were also asked comprehension questions after two-thirds of the sentences. In order to remove effects of processing difficulty due to length, Bach et al. subtracted the ratings for the right-branching paraphrase sentences from the matched complex recursive test sentences.

The same procedure was applied to the error scores from the comprehension questions. The resulting difference should thus reflect the difficulty caused by complex recursion.

Figure 5 (left panel) shows the difference in mean test/paraphrase ratings for singly and doubly embedded cross-dependency sentences in Dutch and German. We focus on the past participle German results because these were consistent across both the rating and comprehension tasks, and were comparable with the Dutch data. Mean GPE across a sentence reflects how difficult the sentence was to process for the SRN. Hence, we can map GPE onto the human sentence rating data, which are thought to reflect the difficulty that subjects experience when processing a given sentence. We used the mean GPEs from Figure 3 for the SRNs trained on center-embeddings and cross-dependencies to model the Bach et al. results. For recursive depth 1 and 2, mean GPEs for the right-branching constructions were subtracted from the average GPEs for the complex recursive constructions, and the differences plotted in Figure 5 (right panel).[13] The net trained on cross-dependencies maps onto the Dutch data and the net trained on center-embedding maps onto the German (past participle) data. At a single level of embedding, Bach et al. found no difference between Dutch and German, and this holds in the SRN data ($t(18) = 0.36, p > 0.7$). However, at two levels of embedding Bach et al. found that Dutch cross-dependency stimuli were rated significantly better than their German counterparts. The SRN data also shows a significant difference between depth 2 center-embeddings and cross-dependencies ($t(18) = 4.08, p < 0.01$). Thus, SRN performance mirrors the human data quite closely.

——————insert figure 5 about here——————

5.4.2 Grammatical vs. ungrammatical double center-embeddings

The study of English sentences with multiple center-embeddings is an important source of information about the limits of human sentence processing (e.g., Blaubergs & Braine, 1974; Foss & Cairns, 1970; Marks, 1968; Miller, 1962; Miller & Isard, 1964; Stolz, 1967). A particularly interesting recent finding (Gibson and Thomas, 1999), using an off-line rating task, suggests that some ungrammatical sentences involving doubly center-embedded object relative clauses may be perceived as grammatical.

(17) *The apartment that the maid who the service had sent over was cleaning every week was well decorated.*

(18) * *The apartment that the maid who the service had sent over was well decorated.*

In particular, they found that when the middle VP was removed (as in 18), the result was rated no worse than the grammatical version (in 17).

Turning to the SRN, in the artificial center embedding language, (17) corresponds to 'NNNVVV', whereas the (18) corresponds to ('NNNVV'). Does the output activation following 'NNNVV' fit the Gibson and Thomas data? Figure 6 shows mean activation across 10 novel sentences and grouped into the four lexical categories and EOS marker. In contrast to the results of Gibson and Thomas, the network demonstrated a significant preference for the ungrammatical 2VP construction over the grammatical 3VP construction, predicting that (17) should be rated worse than (18).

————insert figure 6 about here————

Gibson and Thomas (1999) employed an off-line task, which might explain why (17) was rated worse than (18). Christiansen and MacDonald (2000) conducted an on-line self-paced word-by-word (center presentation) grammaticality judgment task using Gibson and Thomas' stimuli. At each point in a sentence subjects judged whether what they had read was a grammatical sentence or not. Following each sentence (whether accepted or rejected), subjects rated the sentences on a 7-point scale (1 = good, 7 = bad). Christiansen and MacDonald found that the grammatical 3VP construction was again rated significantly worse than the ungrammatical 2VP construction.

One potential problem with this experiment is that the 2VP and 3VP stimuli were different lengths, introducing a possible confound. The Gibson and Thomas stimuli also incorporated semantic biases (e.g., *apartment/decorated, maid/cleaning, service/sent over* in (17)) which may make the 2VP stimuli seem spuriously plausible. Christiansen and MacDonald therefore replicated their first experiment using stimuli controlled for length and without noun/verb biases, such as (19) and (20):

(19) *The chef who the waiter who the busboy offended appreciated admired the musicians.*

(20) * *The chef who the waiter who the busboy offended frequently admired the musicians.*

Figure 7 shows the rating from the second experiment in comparison with SRN mean GPEs. As before, Christiansen and MacDonald found that grammatical 3VP constructions were rated as significantly worse than the ungrammatical 2VP constructions. The SRN data fitted this pattern with significantly higher GPEs in 3VP constructions compared with 2VP constructions ($t(18) = 2.34, p < 0.04$).

——————insert figure 7 about here——————

5.4.3 Right-branching subject relative constructions

Traditional symbolic models suggest that right-branching recursion should not cause processing problems. In contrast, we have seen that the SRN shows some decrement with increasing recursion depth. This issue has received little empirical attention. However, right-branching constructions are often control items in studies of center-embedding, and some relevant information can be gleaned from some of these studies. For example, Bach et al. (1986) report comprehensibility ratings for their right-branching paraphrase items. Figure 8 shows the comprehensibility ratings for the German past participle paraphrase sentences as a function of recursion depth, and mean SRN GPEs for right-branching constructions (from Figure 3) for the center-embedding language. Both the human and the SRN data show the same pattern of increasing processing difficulty with increasing recursion depth.

————insert figure 8 about here————

A similar fit with human data is found by comparing the human comprehension errors as a function of recursion depth reported in Blaubergs and Braine (1974) with mean GPE for the same depths of recursion (again for the SRN trained on the center-embedding language). Christiansen and MacDonald (2000) present on-line rating data concerning right-branching PP modifications of nouns in which the depth of recursion varied from 0 to 2 by modifying a noun by either one PP (21), two PPs (22), or three PPs (23):

(21) *The nurse with the vase says that the [flowers by the window] resemble roses.*

(22) *The nurse says that the [flowers in the vase by the window] resemble roses.*

(23) *The blooming [flowers in the vase on the table by the window] resemble roses.*

The stimuli were controlled for length and propositional and syntactic complexity. The results showed that subjects rated sentences with recursion of depth 2 (23) worse than sentences with recursion depth 1 (22), which, in turn, were rated worse than sentences with no recursion (21). Although these results do not concern subject relative constructions, they suggest that processing right-branching recursive constructions is affected by recursion depth—although the effect of increasing depth is less severe than in complex recursive constructions. Importantly, this dovetails with the SRN predictions (Christiansen, 1994, 2000; Christiansen and MacDonald, 2000), though not with symbolic models of language processing (e.g., Church, 1982; Gibson, 1998; Marcus, 1980; Stabler, 1994).

5.4.4 Counting recursion

Finally, we briefly discuss the relationship between counting recursion and natural language. We contend that, despite Chomsky (1957), such structures may not exist in natural language. Indeed, the kind of structures that Chomsky had in mind (e.g., nested *'if–then'* structures) seem closer to center-embedded constructions than to counting recursive structures. Consider the earlier mentioned depth 1 example (16), repeated here as (24):

(24) *If*$_1$ *if*$_2$ *the cat is in, then*$_2$ *the dog cannot come in then*$_1$ *the cat and dog dislike each other.*

As the subscripts indicate, the *'if–then'* pairs are nested in a center-embedding order. This structural ordering becomes even more evident when we mix *'if–then'* pairs with *'either–or'* pairs (as suggested by Chomsky, 1957: p. 22):

(25) *If*$_1$ *either*$_2$ *the cat dislikes the dog, or*$_2$ *the dog dislikes the cat then*$_1$ *the dog cannot come in.*

(26) *If*$_1$ *either*$_2$ *the cat dislikes the dog, then*$_1$ *the dog dislikes the cat or*$_2$ *the dog cannot come in.*

The center-embedding ordering seems necessary in (25) because if we reverse the order of *'or'* and *'then'* then we get the obscure sentence in (26). Thus, we predict that human behavior on nested *'if–then'* structures should follow the same breakdown pattern as for nested center-embedded constructions (perhaps with a slightly better overall performance).

5.5 Probing the Internal Representations

We now consider the basis of SRN performance by analyzing the HU representations with which the SRNs store information about previous linguistic material. We focus on the doubly embedded constructions, which represent the limits of performance for both people and the SRN. Moreover, we focus on what information the SRN's HUs maintain about the number agreement of the three nouns encountered in doubly embedded constructions (recording the HUs' activations immediately after the three nouns have been presented).

We first provide an intuitive motivation for our approach. Suppose that we aim to assess how much information the HUs maintain about the number agreement of the last noun in a sentence; that is, the noun that the net has just seen. If the information is maintained well, then the HU representations of input sequences that end with a singular noun (and thus belong to the lexical category combinations: nn-**n**, nN-**n**, Nn-**n** and NN-**n**) will be well-separated in HU space from the representations of the input sequences ending in a plural noun (i.e., NN-**N**, Nn-**N**, nN-**N** and nn-**N**). Thus, it should be possible to split the HU representations *along* the plural/singular noun category boundary such that inputs ending in plural nouns are separated from inputs ending in singular nouns. It is important to contrast this with a situation in which the HU representations instead retain information about the agreement number of individual nouns. In this case, we should be able to split the HU representations *across* the plural/singular noun category boundary such that input sequences ending with particular nouns, say, N_1, n_1, N_2 or n_2 (i.e., nn-$\{N_1, n_1, N_2, n_2\}$,[14] nN-$\{N_1, n_1, N_2, n_2\}$, Nn-$\{N_1, n_1, N_2, n_2\}$ and NN-$\{N_1, n_1, N_2, n_2\}$) are separated from inputs ending with remaining nouns N_3, n_3, N_4 or n_4 (i.e., nn-$\{N_3, n_3, N_4, n_4\}$, nN-$\{N_3, n_3, N_4, n_4\}$, Nn-$\{N_3, n_3, N_4, n_4\}$ and NN-$\{N_3, n_3, N_4, n_4\}$). Note that the above separation along lexical categories is a special case of across category separation in which inputs ending with the particular (singular) nouns n_1, n_2, n_3 or n_4 are separated from input sequences ending with the remaining (plural) nouns N_1, N_2, N_3 or N_4. Only by comparing the separation along and across the lexical categories of singular/plural nouns can we assess whether the HU representations merely maintain agreement information about individual nouns, or whether more abstract knowledge has been encoded pertaining to the categories of singular and plural nouns. In both cases, information is maintained relevant to the prediction of correctly agreeing verbs, but only in the latter case are such predictions based on a generalization from the occurrences of individual nouns to their respective categories of singular and plural nouns.

We can measure the degree of separation by attempting to split the HU representations generated from the ($8 \times 8 \times 8 =$) 512 possible sequences of three nouns into two equal groups. We attempt to make this split using a plane in HU space; the degree to which two groups can be separated either along or across lexical categories therefore provides a measure of what information the network maintains about the number agreement of the last noun. A standard statistical test for the separability of two groups of items is discriminant analysis (Cliff, 1987; see Bullinaria, 1994; Wiles & Bloesch, 1992; Wiles & Ollila, 1993 for earlier applications to connectionist networks).

Figure 9(a) schematic illustrates a separation along lexical categories with a perfect differentiation of the two groups, corresponding to a 100% correct vector classification. The same procedure can be used to assess the amount of information that the HUs maintain concerning the number agreement of the nouns in second and first positions. We split the same HU activations generated from the 512 possible input sequences into groups both along and across lexical categories. The separation of the HU vectors along the lexical categories according to the number of the second noun in Figure 9(b) is also perfect. However, as illustrated by Figure 9(c), the separation of the HU activations along the lexical categories according to the first encountered noun is less good, with 75% of the vectors correctly classified, because **N**-Nn is incorrectly classified with the singulars and **n**-nN with the plurals.

——————insert figure 9 about here——————

We recorded HU activations for the 512 possible noun combinations for complex and right-branching recursive constructions of depth 2 (ignoring the interleaving verbs in the right-branching structures). Table 3 lists the percentage of correctly classified HU activations for each combination. Classification scores were found for these combinations both before and after training, and both for separation along and across singular/plural noun categories. Scores were averaged over different initial weight configurations and collapsed across the SRNs trained on the three languages (there were no significant differences between individual scores). The results from the separations across singular/plural noun categories show that prior to any training the SRN retained a considerable amount of information about the agreement number of individual nouns in the last and middle positions. Only for the first encountered noun was performance essentially at chance (i.e., close to the performance achieved through a random assignment of the vectors into two groups). The SRN had, not surprisingly, no knowledge of lexical categories of singular and plural nouns before training, as indicated by the lack of difference between the classification scores along and across noun categories. The good classification performance of the untrained nets on the middle noun in the right-branching constructions is, however, somewhat surprising because this noun position is two words (a verb and a noun) away from the last noun. In terms of absolute position from the point where the HU activations were recorded, the middle noun in right-branching constructions (e.g., ‘$N_1V_3 - \mathbf{N_3} - V_2n_4$’) corresponds to the first noun in complex recursive constructions (e.g., ‘$\mathbf{N_1} - N_3n_4$’). Whereas untrained classification performance for this position was near chance on complex recursion, it was near perfect on right-branching recursion. This suggests that in the latter case information about the verb, which occurs between the last and the middle nouns, does not interfere much with the retention of agreement information about the middle noun. Thus, prior to learning, the SRN appears to have an architectural bias which facilitates processing right-branching structures over complex recursive structures.

——————insert table 3 about here——————

After training, the SRN HUs retained less information about individual nouns. Instead, lexical category information was maintained as evidenced by the big differences in classification scores between groups separated along and across singular/plural noun categories. Whereas classification scores along the two noun categories increased considerably as a result of training, the scores for classifications made according to groups separated across the categories of singular and plural nouns actually decreased—especially for the middle noun position. The SRN appears to have learned about the importance of the lexical categories of singular and plural nouns for the purpose of successful performance on the prediction task, but at the cost of losing information about individual nouns in the middle position.

The results of the discriminant analyses suggest that the SRN is well-suited for learning sequential dependencies. The feedback between the context layer and the hidden layer allows the net to retain information relevant to appropriate distinctions between previously encountered plural and singular items even prior to learning. Of course, a net has to learn to take advantage of this initial separation of the HU activations to produce the correct output, which is a nontrivial task. Prior to learning, the output of an SRN consist of random activation patterns. Thus, it must discover the lexical categories and learn to apply agreement information in the right order to make correct predictions for center-embeddings and cross-dependencies.

On a methodological level, these results suggest that analyses of the untrained networks should be used as baselines for analyses of HU representations in trained networks. This may provide insight into which aspects of network performance are due to architectural biases and which arise from learning. A network always has some bias with respect to a particular task, and this bias depends on several factors, such as overall network configuration, choice of activation function, choice of input/output representations, initial weight setting, etc. As evidenced by our discriminant analyses, even prior to learning, HU representations may display some structural differentiation, emerging as the combined product of this bias (also cf. Kolen, 1994) and the statistics of the input/output relations in the test material. However, all too often HU analyses—such as cluster analyses, multi-dimensional scaling analyses, principal component analyses—are conducted without any baseline analysis of untrained networks.

6 General Discussion

We have shown that SRNs can learn to process recursive structures with similar performance limitations regarding depth of recursion as in human language processing. The SRNs limitations appear relatively insensitive to the size of the network and the frequency of deeply recursive structures in the training input. The qualitative pattern of SRN results match human performance on natural language constructions with these structures. The SRNs trained on center-embedding and cross-dependency constructions performed well on singly embedded sentences—although, as for people, performance was by no means perfect (Bach et al., 1986; Blaubergs & Braine, 1974; King & Just, 1991). Of particular interest is the pattern of performance degradation on sentences involving center-embeddings and cross-dependencies of depth 2, and its close match with the pattern of human performance.

These encouraging results suggest a reevaluation of Chomsky's (1957, 1959) arguments that the existence of recursive structures in language rules out finite state and associative models of language processing. These arguments have been taken to indicate that connectionist networks cannot in principle account for human language processing. But we have shown that this in-principle argument is not correct. Connectionist networks can learn to handle recursion with a comparable level of performance to people. Our simulations are, of course, small scale, and do not show that this approach generalizes to model the acquisition of the full complexity of natural language. But this limitation applies equally well to symbolic approaches to language acquisition (e.g., Anderson, 1983), including parameter-setting models (e.g., Gibson & Wexler, 1994; Niyogi & Berwick, 1996), and other models which assume an innate universal grammar (e.g., Berwick & Weinberg, 1984).

Turning to linguistic issues, the better SRN performance on cross-dependencies over center-embeddings may reflect the fact that the problem of learning limited versions of context-free and context-sensitive languages may be very different from the problem of learning the full, infinite versions of these languages (compare Vogel, Hahn and Branigan, 1996). Within the framework of Gibson's (1998) Syntactic Prediction Locality Theory, center-embedded constructions (of depth 2 or less) are harder to process than their cross-dependency counterparts because center-embedding requires holding information in memory over a longer stretch of intervening items. Although a similar explanation is helpful in understanding the difference in SRN performance on the two types of complex recursive constructions, this cannot be the full explanation. Firstly, this analysis incorrectly suggests that singly embedded cross-dependency structures should be easier than comparable center-embedded constructions. As illustrated by Figure 5, this is not true of the SRN predictions, nor in the human data from Bach et al. (1986). Secondly, the above analysis predicts a flat or slightly rising pattern of GPE across the verbs in a sentence with two cross-dependencies. In contrast, the GPE pattern for the cross-dependency sentences (Figure 4) fits the reading time data from Dickey and Vonk (1997) because of a *drop* in the GPEs for the second verb. Overall, the current results suggest that we should be wary of drawing strong conclusions for language processing, in networks and perhaps also in people, from arguments concerning idealized infinite cases.

A related point concerns the architectural requirements for learning languages involving, respectively, context-free and context-sensitive structures. In our simulations, the very same network learned the three different artificial languages to a degree similar to human performance. To our knowledge, no symbolic model has been shown to be able to *learn* these three kinds of recursive structures given *identical initial conditions*. For example, Berwick and Weinberg's (1984) symbolic model of language acquisition has a built-in stack and would therefore not be able to process cross-dependencies. Of course, if one builds a context-sensitive parser then it can also by definition parse context-free strings. However, the processing models that are able to account for the Bach et al. (1986) data (Gibson, 1998; Joshi, 1990; Rambow & Joshi, 1994) do not incorporate theories of learning that can explain how the ability to process center-embedding and cross-dependency could be acquired.

In this chapter, we have presented results showing a close qualitative similarity between breakdowns in human and SRN processing when faced with complex recursion. This was achieved without assuming that the language processor has access to a competence grammar which allows unbounded recursion, subject to performance constraints. Instead, the SRN account suggests that the recursive constructions that people actually say and hear may be explained by a system with no representation of unbounded grammatical competence, and performance limitations arise from intrinsic constraints on processing. If this hypothesis is correct, then the standard distinction between competence and performance, which is at the center of contemporary linguistics, may need to be rethought.

Further Readings

Most of the early connectionist models of recursion were essentially simple re-implementations of symbolic parsers (e.g., Fanty, 1986; Small, Cottrell & Shastri, 1982). The first more comprehensive model of this kind was McClelland and Kawamoto's (1986) neural network model of case-role assignment. Many of the subsequent models of sentence processing and recursion have sought to provide alternatives to the symbolic processing models. One approach has been to learn recursive structure from "tagged" input sentences. Among these, Pollack's (1988, 1990) recursive auto-associative memory network has inspired several subsequent modeling efforts (e.g., Chalmers, 1990; Niklasson & van Gelder, 1994; see also Steedman, Chapter 11, this volume). Another approach is to construct a modular system of networks, each of which is trained to acquire different aspects of syntactic processing. Miikkulainen's (1996) three-network system provides a good example of this approach. But the most popular connectionist approach to recursion and syntactic processing builds on Elman's (1990, 1991, 1993) Simple Recurrent Network model.

Recently, efforts have been made to model reading time data from recursive sentence processing experiments. The work by Christiansen (2000; Christiansen & Chater, 1999; MacDonald & Christiansen, in press) is perhaps the best example of this line of research. Turning to syntactic processing more generally, Tabor, Juliano & Tanenhaus (1997) provide a dynamical sentence processing model (see also, Tabor & Tanenhaus, Chapter 6, this volume). The most influential non-connectionist model of sentence processing results is Gibson's (1998) Syntactic Prediction Locality Theory model. A slightly older non-connectionist model is the CC-READER model by Just and Carpenter (1992).

For discussions of the future prospects of connectionist models of syntax (and recursion), see Seidenberg and MacDonald (Chapter 9, this volume) and Steedman (Chapter 11, this volume).

References

Anderson, J.R. (1983). *The architecture of cognition.* Cambridge, MA: Harvard University Press.

Bach, E., Brown, C. & Marslen-Wilson, W. (1986). Crossed and nested dependencies in German and Dutch: A psycholinguistic study. *Language and Cognitive Processes*, *1*, 249–262.

Batali, J. (1994). Artificial evolution of syntactic aptitude. In *Proceedings from the Sixteenth Annual Conference of the Cognitive Science Society* (pp. 27–32). Hillsdale, NJ: Lawrence Erlbaum.

Berwick, R.C & Weinberg, A.S (1984). *The grammatical basis of linguistic performance: Language use and acquisition.* Cambridge, MA: MIT Press.

Blaubergs, M.S. & Braine, M.D.S. (1974). Short-term memory limitations on decoding self-embedded sentences. *Journal of Experimental Psychology*, *102*, 745–748.

Bullinaria, J.A. (1994). Internal representations of a connectionist model of reading aloud. In *Proceedings from the Sixteenth Annual Conference of the Cognitive Science Society* (pp. 84–89). Hillsdale, NJ: Lawrence Erlbaum.

Chalmers, D.J. (1990). Syntactic transformations on distributed representations. *Connection Science*, *2*, 53–62.

Charniak, E. (1993). *Statistical language learning.* Cambridge, MA: MIT Press.

Chater, N. & Conkey, P. (1992). Finding linguistic structure with recurrent neural networks. In *Proceedings of the Fourteenth Annual Meeting of the Cognitive Science Society* (pp. 402–407). Hillsdale, NJ: Lawrence Erlbaum.

Chomsky, N. (1957). *Syntactic structures.* The Hague: Mouton.

Chomsky, N. (1959). Review of Skinner (1957). *Language*, *35*, 26–58.

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

Christiansen, M.H. (1992). The (non)necessity of recursion in natural language processing. In *Proceedings of the Fourteenth Annual Meeting of the Cognitive Science Society* (pp. 665–670). Hillsdale, NJ: Lawrence Erlbaum.

Christiansen, M.H. (1994). *Infinite languages, finite minds: Connectionism, learning and linguistic structure.* Unpublished doctoral dissertation, University of Edinburgh.

Christiansen, M.H. (2000). *Intrinsic constraints on the processing of recursive sentence structure.* Manuscript in preparation,

Christiansen,M.H. & Chater, N. (1994). Generalization and connectionist language learning. *Mind and Language*, *9*, 273–287.

Christiansen, M.H. & Chater, N. (1999). Toward a connectionist model of recursion in human linguistic performance. *Cognitive Science*, *23*, 157–205.

Christiansen, M.H. & MacDonald, M.C. (2000). *Processing of recursive sentence structure: Testing predictions from a connectionist model.* Manuscript in preparation,

Church, K. (1982). *On memory limitations in natural language processing.* Bloomington, IN: Indiana University Linguistics Club.

Cleeremans, A., Servan-Schreiber, D. & McClelland, J. L. (1989). Finite state automata and simple recurrent networks. *Neural Computation*, *1*, 372–381.

Cliff, N. (1987). *Analyzing multivariate data.* Orlando, FL: Harcourt Brace Jovanovich.

Dickey, M.W. & Vonk, W. (1997). Center-embedded structures in Dutch: An on-line study. Poster presented at the Tenth Annual CUNY Conference on Human Sentence Processing. Santa Monica, CA, March 20–22.

Elman, J.L. (1990). Finding structure in time. *Cognitive Science*, *14*, 179–211.

Elman, J.L. (1991). Distributed representation, simple recurrent networks, and grammatical structure. *Machine Learning*, *7*, 195–225.

Elman, J.L. (1993). Learning and development in neural networks: The importance of starting small. *Cognition*, *48*, 71–99.

Fanty, M. (1986). Context-free parsing with connectionist networks. In J.S. Denker (Ed.), *Neural networks for computing* (AIP Conference Proceedings 151) (pp. 140–145). New York: American Institute of Physics.

(Tech. Rep. No. TR-174). Rochester, NY: University of Rochester, Department of Computer Science.

Foss, D.J. & H.S. Cairns (1970). Some effects of memory limitations upon sentence comprehension and recall. *Journal of Verbal Learning and Verbal Behavior*, *9*, 541–547.

Gale, W. & Church, K. (1990). Poor estimates of context are worse than none. In *Proceedings of the June 1990 DARPA Speech and Natural Language Workshop*. Hidden Valley, PA.

Gazdar, G. & Pullum, G.K. (1985). *Computationally relevant properties of natural languages and their grammars* (Tech. Rep. No. CSLI-85-24). Palo Alto, CA: Stanford University, Center for the Study of Language and Information.

Gibson, E. (1998). Linguistic complexity: Locality of syntactic dependencies. *Cognition*, *68*, 1–76.

Gibson, E. & Thomas, J. (1999). Memory limitations and structural forgetting: The perception of complex ungrammatical sentences as grammatical. *Language and Cognitive Processes*, *14*, 225–248.

Gibson, E. & Wexler, K. (1994). Triggers. *Linguistic Inquiry*, *25*, 407–454.

Giles, C. & Omlin, C. (1993). Extraction, insertion and refinement of symbolic rules in dynamically driven recurrent neural networks. *Connection Science*, *5*, 307–337.

Giles, C., Miller, C., Chen, D., Chen, H., Sun, G., & Lee, Y. (1992). Learning and extracting finite state automata with second-order recurrent neural networks. *Neural Computation*, *4*, 393–405.

Green, D.M. & Swets, J.A. (1966). *Signal detection theory and psychophysics.* New York: Wiley.

Hanson, S.J. & Kegl, J. (1987). PARSNIP: A connectionist network that learns natural language grammar from exposure to natural language sentences. In *Proceedings of the Eight Annual Meeting of the Cognitive Science Society* (pp. 106–119). Hillsdale, NJ: Lawrence Erlbaum.

Joshi, A.K. (1990). Processing crossed and nested dependencies: An automaton perspective on the psycholinguistic results. *Language and Cognitive Processes*, *5*, 1–27.

Just, M.A. & Carpenter, P.A. (1992). A capacity theory of comprehension: Individual differences in working memory. *Psychological Review*, *99*, 122-149.

King, J. & Just, M.A. (1991). Individual differences in syntactic processing: The role of working memory. *Journal of Memory and Language*, *30*, 580–602.

Kolen, J.F. (1994). The origin of clusters in recurrent neural network state space. In *Proceedings from the Sixteenth Annual Conference of the Cognitive Science Society* (pp. 508–513). Hillsdale, NJ: Lawrence Erlbaum.

Larkin, W. & Burns, D. (1977). Sentence comprehension and memory for embedded structure. *Memory & Cognition*, **5**, 17–22.

Luce, D. (1959). *Individual choice behavior.* New York: Wiley.

McClelland, J.L. & Kawamoto, A.H. (1986). Mechanisms of sentence processing. In J.L. McClelland & D.E. Rumelhart (Eds.), *Parallel Distributed Processing, Volume 2* (pp. 272–325). Cambridge, MA.: MIT Press.

MacDonald, M.C. & Christiansen, M.H. (in press). Reassessing working memory: A comment on Just & Carpenter (1992) and Waters & Caplan (1996). *Psychological Review.*

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

Marks, L.E. (1968). Scaling of grammaticalness of self-embedded English sentences. *Journal of Verbal Learning and Verbal Behavior*, *7*, 965–967.

Miikkulainen, R. (1996). Subsymbolic case-role analysis of sentences with embedded clauses. *Cognitive Science*, **20**, 47–73.

Miller, G.A. (1962). Some psychological studies of grammar. *American Psychologist*, *17*, 748–762.

Miller, G.A. & Isard, S. (1964). Free recall of self-embedded English sentences. *Information and Control*, *7*, 292–303.

Niklasson, L. & van Gelder (1994). On being systematically connectionist. *Mind and Language*, *9*, 288–302.

Niyogi, P. & Berwick, R.C. (1996). A language learning model for finite parameter spaces. *Cognition*, *61*, 161–193.

Pollack, J.B. (1988). Recursive auto-associative memory: Devising compositional distributed representations. In *Proceedings of the Tenth Annual Meeting of the Cognitive Science Society* (pp. 33–39). Hillsdale, NJ: Lawrence Erlbaum.

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

Pullum, G.K. & Gazdar, G. (1982). Natural languages and context-free languages. *Linguistics and Philosophy*, *4*, 471–504.

Rambow, O. & Joshi, A.K. (1994). A processing model for free word-order languages. In C. Clifton, L. Frazier & K. Rayner (Eds.), *Perspectives on sentence processing* (pp. 267–301). Hillsdale, NJ: Lawrence Erlbaum.

Redington, M., Chater, N. & Finch, S. (1998). Distributional information: A powerful cue for acquiring syntactic categories. *Cognitive Science*, *22*, 425–469.

Reich, P. (1969). The finiteness of natural language. *Language*, *45*, 831–843.

Rumelhart, D.E., Hinton, G.E. & Williams, R.J. (1986). *Learning internal representations by error propagation.* In McClelland, J.L. & Rumelhart, D.E. (Eds.) *Parallel distributed processing, Vol. 1.* (pp. 318–362). Cambridge, MA: MIT Press.

Schütze, C.T. (1996). *The empirical base of linguistics: Grammaticality judgments and linguistic methodology.* Chicago, IL: The University of Chicago Press.

Servan-Schreiber, D., Cleeremans, A. & McClelland, J. L. (1991). Graded state machines: The representation of temporal contingencies in simple recurrent networks. *Machine Learning*, *7*, 161–193.

Shieber, S. (1985). Evidence against the context-freeness of natural language. *Linguistics and Philosophy*, *8*, 333–343.

Skinner, B.F. (1957). *Verbal Behavior.* New York: Appleton-Century-Crofts.

Small, S.L., Cottrell, G.W. & Shastri, L. (1982). Towards connectionist parsing. In *Proceedings of the National Conference on Artificial Intelligence.* Pittsburgh, PA.

Stabler, E.P. (1994). The finite connectivity of linguistic structure. In C. Clifton, L. Frazier & K. Rayner (Eds.), *Perspectives on sentence processing* (pp. 303–336). Hillsdale, NJ: Lawrence Erlbaum.

Steijvers, M. & Grünwald, P. (1996). A recurrent network that performs a context-sensitive prediction task. In *Proceedings from the Eighteenth Annual Conference of the Cognitive Science Society* (pp. 335–339). Mahwah, NJ: Lawrence Erlbaum.

Stolcke, A. (1991). Syntactic category formation with vector space grammars. In *Proceedings from the Thirteenth Annual Conference of the Cognitive Science Society* (pp. 908–912). Hillsdale, NJ: Lawrence Erlbaum.

Stolz, W.S. (1967). A study of the ability to decode grammatically novel sentences. *Journal of Verbal Learning and Verbal Behavior*, *6*, 867–873.

Tabor, W., Juliano, C. & Tanenhaus, M.K. (1997). Parsing in a dynamical system: An attractor-based account of the interaction of lexical and structural constraints in sentence processing. *Language and Cognitive Processes*, *12*, 211–271.

Vogel, C., Hahn, U. & Branigan, H. (1996). Cross-serial dependencies are not hard to process. In *Proceedings of COLING-96, The 16th International Conference on Computational Linguistics* (pp. 157–162), Copenhagen, Denmark.

Weckerly, J. & Elman, J. (1992). A PDP approach to processing center-embedded sentences. In *Proceedings of the Fourteenth Annual Meeting of the Cognitive Science Society* (pp. 414–419). Hillsdale, NJ: Lawrence Erlbaum.

Wiles, J. & Bloesch, A. (1992). Operators and curried functions: Training and analysis of simple recurrent networks. In J.E. Moody, S.J. Hanson & R.P. Lippmann (Eds.), *Advances in Neural Information Processing Systems 4*. San Mateo, CA: Morgan-Kaufmann.

Wiles, J. & Elman, J. (1995). Learning to count without a counter: A case study of dynamics and activation landscapes in recurrent networks. In *Proceedings of the Seventeenth Annual Meeting of the Cognitive Science Society* (pp. 482–487). Hillsdale, NJ: Lawrence Erlbaum.

Wiles, J. & Ollila, M. (1993). Intersecting regions: The key to combinatorial structure in hidden unit space. In S.J. Hanson, J.D. Cowan & C.L. Giles (Eds.), *Advances in Neural Information Processing Systems 5* (pp. 27–33). San Mateo, CA: Morgan-Kaufmann.

7 Author Notes

This chapter is in large parts based on Christiansen & Chater (1999). We would like to thank Joe Allen, Jim Hoeffner, Mark Seidenberg and Paul Smolensky for discussions and comments on the work presented here.

Notes

[1] We leave aside generalization, which we discuss elsewhere (Christiansen, 1994, 2000; Christiansen & Chater, 1994).

[2] Cross-dependency has also been alleged to be present in "respectively" constructions in English, such as *'Anita$_1$ and the girls$_2$ walks$_1$ and skip$_2$, respectively'.* Church (1982) questions the acceptability of these constructions with two cross-dependencies, and even one, as in this example, seems bizarre.

[3] Pullum & Gazdar (1982) have argued, controversially, that natural language is, nonetheless, context-free (see Gazdar & Pullum, 1985; Shieber, 1985;).

[4] Their nets were trained using back-propagation through time (Rumelhart, Hinton & Williams, 1986)—see below.

[5] Intuition may suggests that higher order n-gram models should outperform simple bigram and trigram models, because they can encode more extended regularities. However, results using text corpora have shown that higher order n-grams provide for poor predictions due to distributional 'undersampling': many higher order n-grams only have one or very few instances, or do not occur at all in a given corpus (Gale & Church, 1990; Redington, Chater & Finch, 1998).

[6] The relation between grammaticality judgments and processing mechanisms is controversial (see Christiansen, 1994; Schütze, 1996).

[7] These simulations used the *Tlearn* simulator available from the Center for Research on Language, University of California, San Diego.

[8] We adopt the convention that 'n' and 'N' corresponds to categories of nouns, 'v' and 'V' to categories of verbs with capitalization indicating plural agreement. The EOS marker is denoted by '#'. Individual word tokens are denoted by adding a subscript, e.g., 'N_3'.

[9] We use bold for random variables.

[10] These assumptions are, of course, very unrealistic of the skewed distribution of word-frequencies in natural language, but are nonetheless used for simplicity.

[11] Note that ''total network activation" is not a possible interpretation, because the difference between the total activation and hit activation (see Equation 4) corresponds to the false alarm activation (see Equation 5).

[12] Could GPE hide a failure to make correct agreement predictions for singly center-embedded sentences, such as *'The man$_1$ the boys$_2$ chase$_2$ likes$_1$ cheese'*? If so, one would expect high agreement error for the two verb predictions in the singly center-embedded (complex depth 1) constructions in Figure 3. Agreement error can be calculated as the percentage of verb activation allocated to verbs which do *not* agree in number with their respective nouns. The agreement error for the first and second verbs was 1.00% and 16.85%, respectively. This level of agreement error is comparable with human performance (Larkin and Burns, 1977).

[13] The human data presented here and below involve three different scales of measurement (i.e., differences in mean test/paraphrase comprehensibility ratings, mean grammaticality ratings from 1-7, and mean comprehensibility ratings from 1-9). It was therefore necessary to adjust the scales for the comparisons with the mean GPEs accordingly.

[14] Curly brackets indicate that any of the nouns may occur in this position, creating the following combinations: nn-N_1, nn-n_1, nn-N_2 and nn-n_2.

TABLE 1
A Recursive Set of Rules for
Right-Branching Relative Clauses

S	→	NP VP
NP	→	N (comp S)
VP	→	V (NP)

Note. S = sentence; NP = noun phrase; VP = verb phrase; N = noun; comp = complementizer; V = verb; Constituents in parentheses are optional.

TABLE 2
The Distribution of Embedding Depths in
Training and Test Corpora

	Embedding Depth			
Recursion Type	0	1	2	3
Complex	15%	27.5%	7%	.5%
Right-Branching	15%	27.5%	7%	.5%
Total	30%	55%	14%	1%

Note. The precise statistics of the individual corpora varied slightly from this ideal distribution.

TABLE 3
Percentage of Cases Correctly Classified given Discriminant Analyses of Network Hidden Unit Representations

	Recursion Type			
	Separation Along Singular/Plural Noun Categories		Separation Across Singular/Plural Noun Categories	
Noun Position	Complex	Right-Branching	Complex	Right-Branching
	Before Training			
First	62.60	52.80	57.62	52.02
Middle	97.92	94.23	89.06	91.80
Last	100.00	100.00	100.00	100.00
Random	56.48	56.19	55.80	55.98
	After Training			
First	96.91	73.34	65.88	64.06
Middle	92.03	98.99	70.83	80.93
Last	99.94	100.00	97.99	97.66
Random	55.99	55.63	54.93	56.11

Notes. Noun position denotes the left-to-right placement of the noun being tested, with Random indicating a random assignment of the vectors into two groups.

Equations

$$P(\mathbf{c_p}|\mathbf{c_1},\mathbf{c_2},\ldots,\mathbf{c_{p-1}}) \simeq \frac{Freq(\mathbf{c_1},\mathbf{c_2},\ldots,\mathbf{c_{p-1}},\mathbf{c_p})}{Freq(\mathbf{c_1},\mathbf{c_2},\ldots,\mathbf{c_{p-1}})} \tag{1}$$

$$P(\mathbf{w_n}|\mathbf{c_1},\mathbf{c_2},\ldots,\mathbf{c_{p-1}}) \simeq \frac{Freq(\mathbf{c_1},\mathbf{c_2},\ldots,\mathbf{c_{p-1}},\mathbf{c_p})}{Freq(\mathbf{c_1},\mathbf{c_2},\ldots,\mathbf{c_{p-1}})\ \mathbf{C_p}} \tag{2}$$

$$\text{Squared Error} = \sum_{j \in W} (out_j - P(\mathbf{w_n} = j))^2 \tag{3}$$

$$\text{hits} = \sum_{i \in G} u_i \tag{4}$$

$$\text{false alarms} = \sum_{i \in U} u_i \tag{5}$$

$$t_i = \frac{(\text{hits + misses})f_i}{\sum_{j \in G} f_j} \tag{6}$$

$$m_i = \begin{cases} 0 & \text{if } t_i - u_i \leq 0 \\ t_i - u_i & \text{otherwise} \end{cases} \tag{7}$$

$$\text{misses} = \sum_{i \in G} m_i \tag{8}$$

$$\text{GPE} = 1 - \frac{\text{hits}}{\text{hits + false alarms + misses}} \tag{9}$$

Figure Captions

Figure 1: The basic architecture of a simple recurrent network (SRN). The rectangles correspond to layers of units. Arrows with solid lines denote trainable weights, whereas the arrow with the dashed line denotes the copy-back connections.

Figure 2: The performance averaged across epochs on complex recursive constructions (left panels) and right-branching constructions (right panels) of nets of different sizes as well as the bigram and trigram models trained on the counting recursion language (top panels), the center-embedding recursion language (middle panels), and the cross-dependency recursion language (bottom panels). Error bars indicate the standard error of the mean.

Figure 3: The mean grammatical prediction error on complex (C) and right-branching (RB) recursive constructions as a function of embedding depth (0-4). Results are shown for the SRN as well as the bigram and trigram models trained on the counting recursion language (top left panel), the center-embedding recursion language (top right panel), and the cross-dependency recursion language (bottom panel).

Figure 4: Grammatical prediction error for each word in doubly embedded sentences for the net trained on constructions of varying length (SRN), the net trained exclusively on doubly embedded constructions (D2-SRN), and the bigram and trigram models. Results are shown for counting recursion (top panel), center-embedding recursion (middle panel), and cross-dependency recursion (bottom panel). Subscripts indicate subject noun/verb agreement patterns.

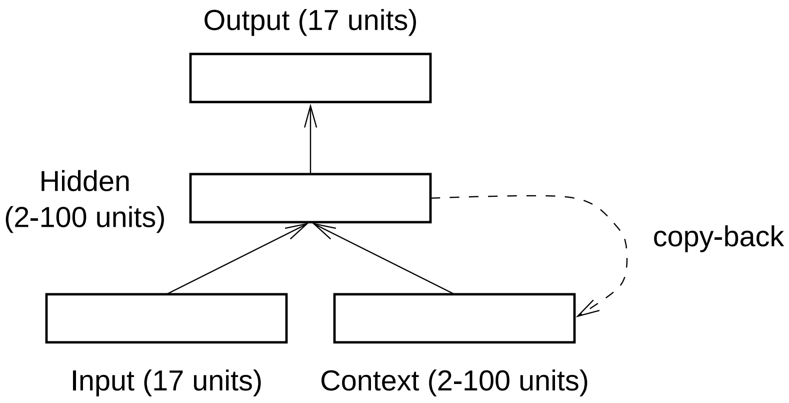
Figure 5: Human performance (from Bach et al., 1986) on singly and doubly center-embedded German (past participle) sentences compared with singly and doubly embedded cross-dependency sentences in Dutch (left panel), and SRN performance on the same kinds of constructions (right panel). Error bars indicate the standard error of the mean.

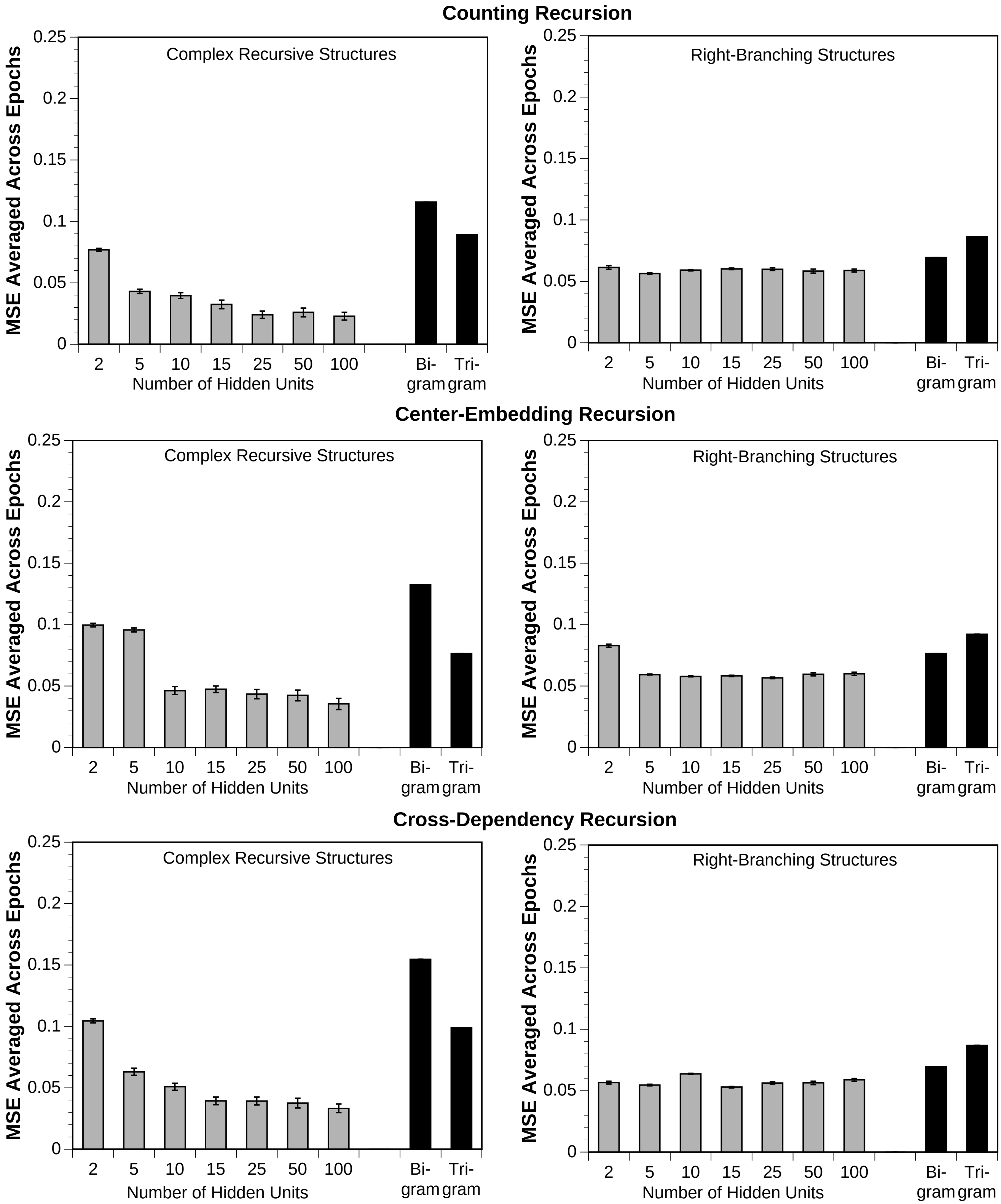
Figure 6: The mean output activation for the four lexical categories and the EOS marker (EOS) given the context 'NNNVV'. Error bars indicate the standard error of the mean.

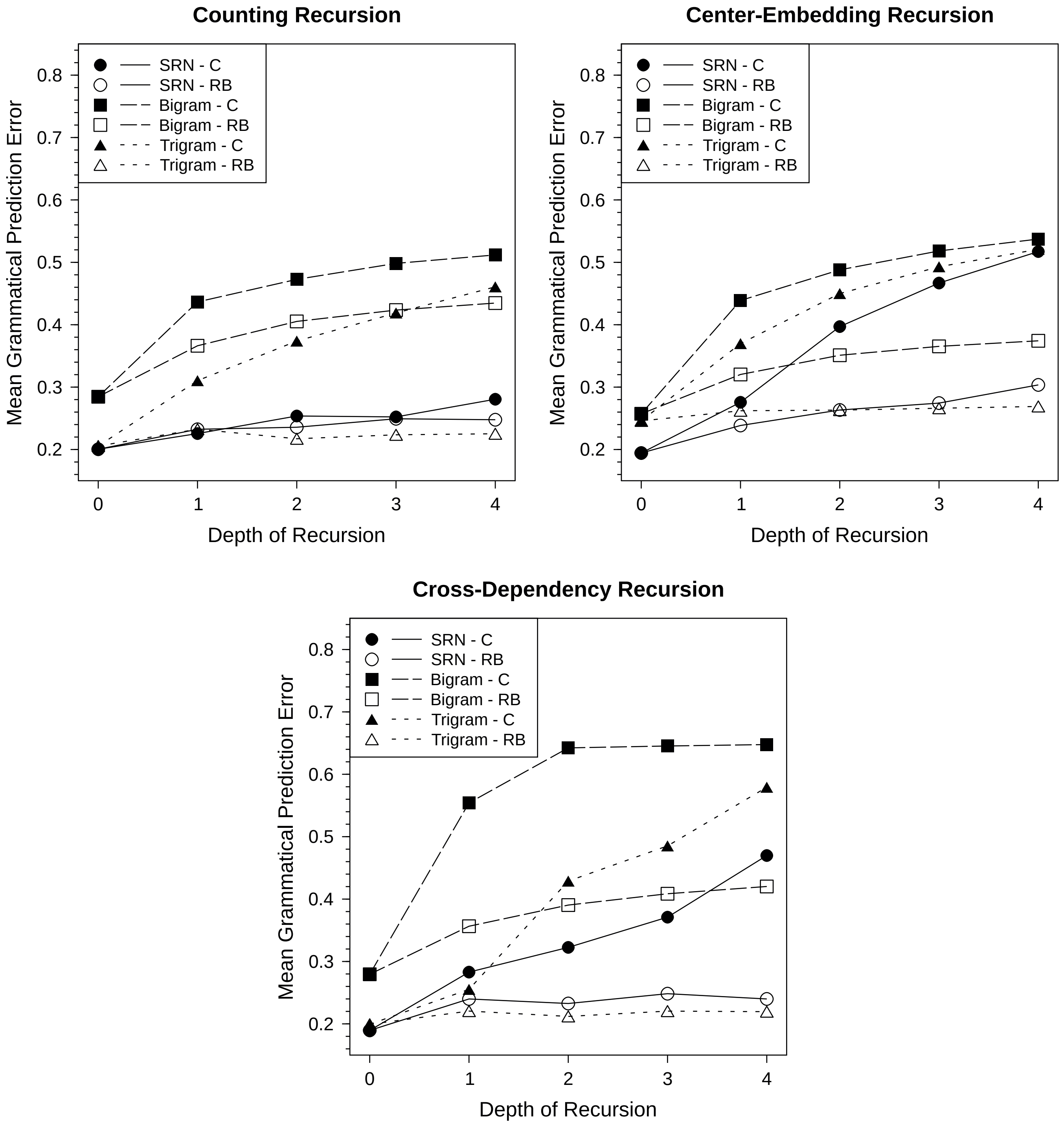
Figure 7: Human ratings (from Christiansen & MacDonald, 2000) for 2VP and 3VP center-embedded English sentences (left ordinate axis) compared with the mean grammatical prediction error produced by the SRN for the same kinds of constructions (right ordinate axis). Error bars indicate the standard error of the mean.

Figure 8: Human comprehensibility ratings (left ordinate axis) from Bach et al. (1996: German past participle paraphrases) compared with the average grammatical prediction error for right-branching constructions produced by the SRN trained on the center-embedding language (right ordinate axis), both plotted as a function of recursion depth.

Figure 9: Schematic illustration of hidden unit state space with each of the noun combinations denoting a cluster of hidden unit vectors recorded for a particular set of agreement patterns (with 'N' corresponding to plural nouns and 'n' to singular nouns). The straight dashed lines represent three linear separations of this hidden unit space according to the number of (a) the last seen noun, (b) the second noun, and (c) the first encountered noun (with incorrectly classified clusters encircled).


Output (17 units)
Hidden
(2-100 units)
copy-back
Input (17 units)
Context (2-100 units)

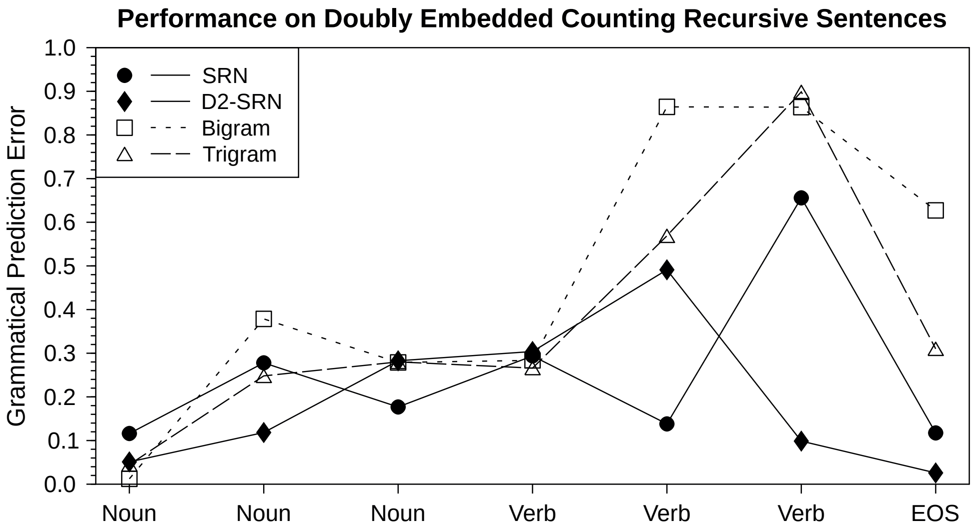

Counting Recursion
Complex Recursive Structures
Right-Branching Structures
Center-Embedding Recursion
Complex Recursive Structures
Right-Branching Structures
Cross-Dependency Recursion
Complex Recursive Structures
Right-Branching Structures
MSE Averaged Across Epochs
0
0.05
0.1
0.15
0.2
0.25
2
5
10
15
25
50
100
Bi-gram
Tri-gram
Number of Hidden Units

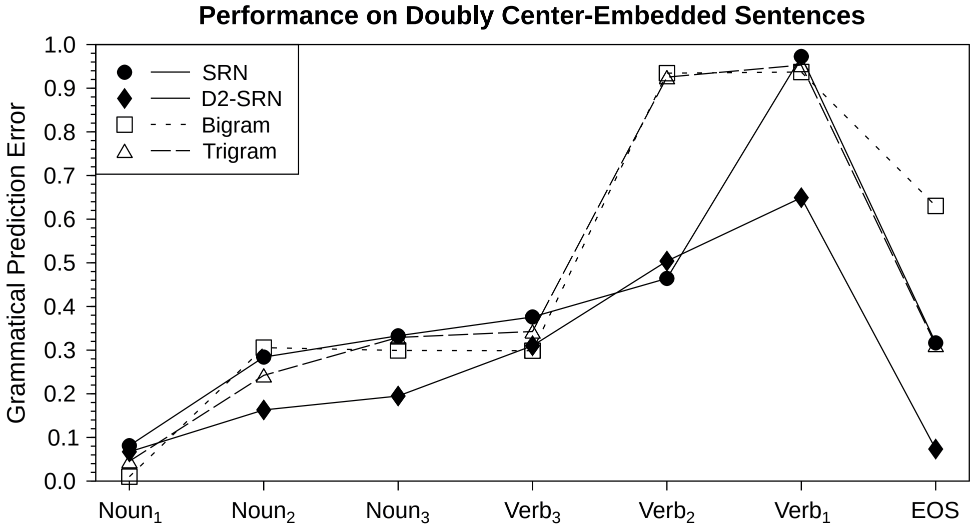


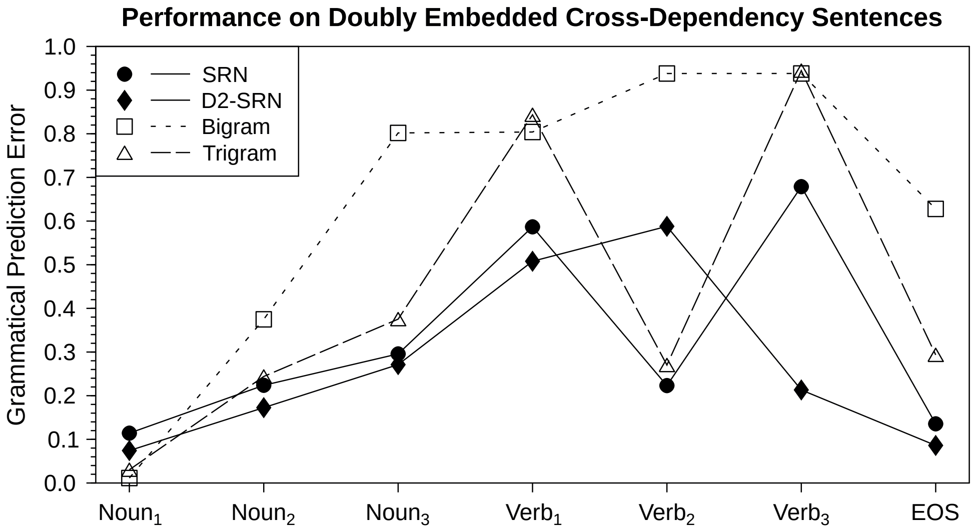
Counting Recursion

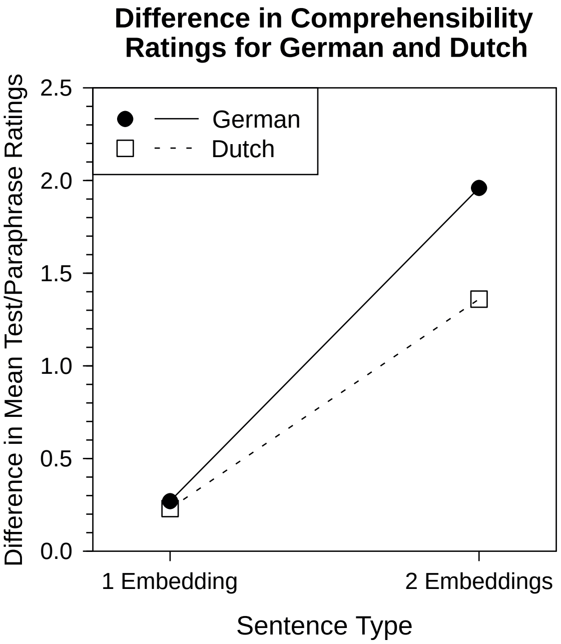
Center-Embedding Recursion

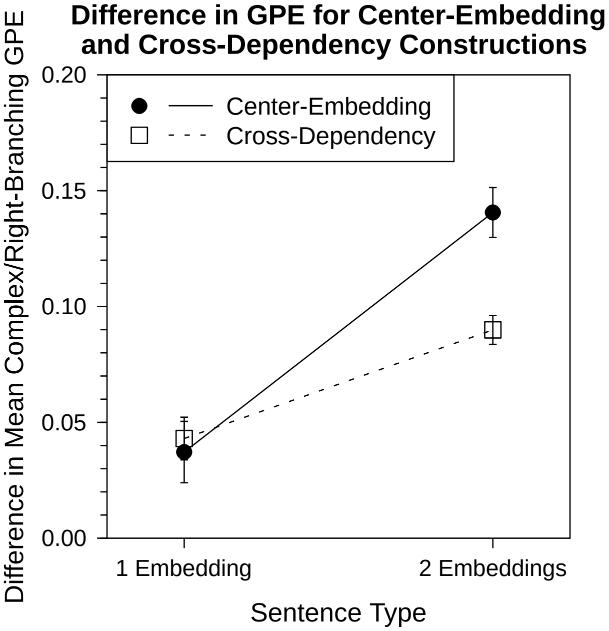
Cross-Dependency Recursion

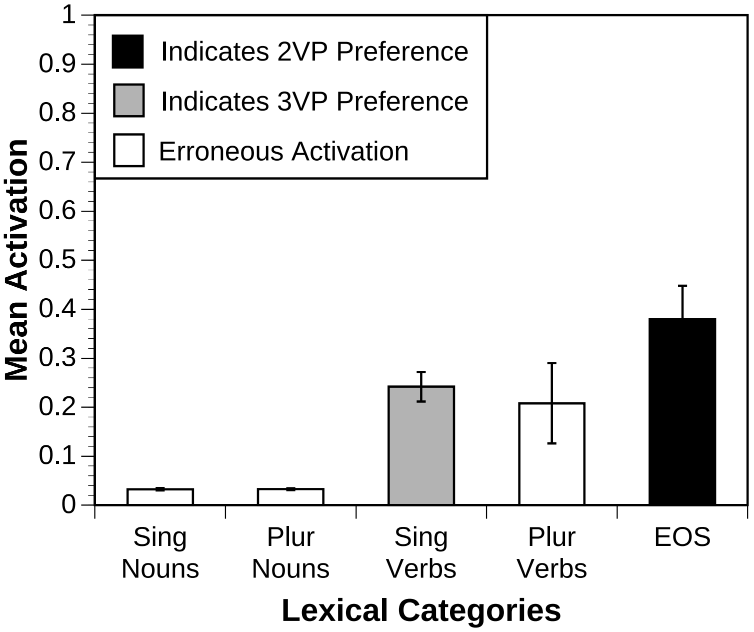

Performance on Doubly Embedded Counting Recursive Sentences
Grammatical Prediction Error
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
SRN
D2-SRN
Bigram
Trigram
Noun
Noun
Noun
Verb
Verb
Verb
EOS

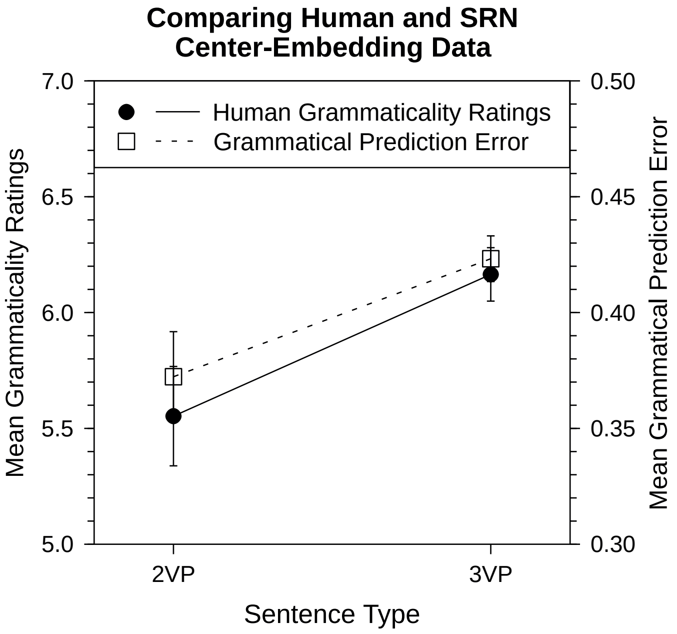

Performance on Doubly Center-Embedded Sentences
Grammatical Prediction Error
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
SRN
D2-SRN
Bigram
Trigram
Noun_1
Noun_2
Noun_3
Verb_3
Verb_2
Verb_1
EOS

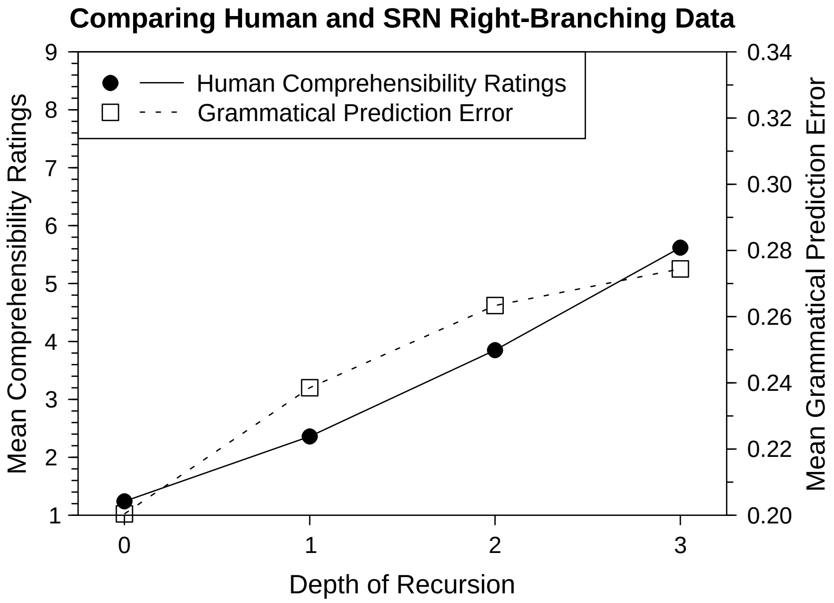

Performance on Doubly Embedded Cross-Dependency Sentences
Grammatical Prediction Error
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
SRN
D2-SRN
Bigram
Trigram
Noun_1
Noun_2
Noun_3
Verb_1
Verb_2
Verb_3
EOS

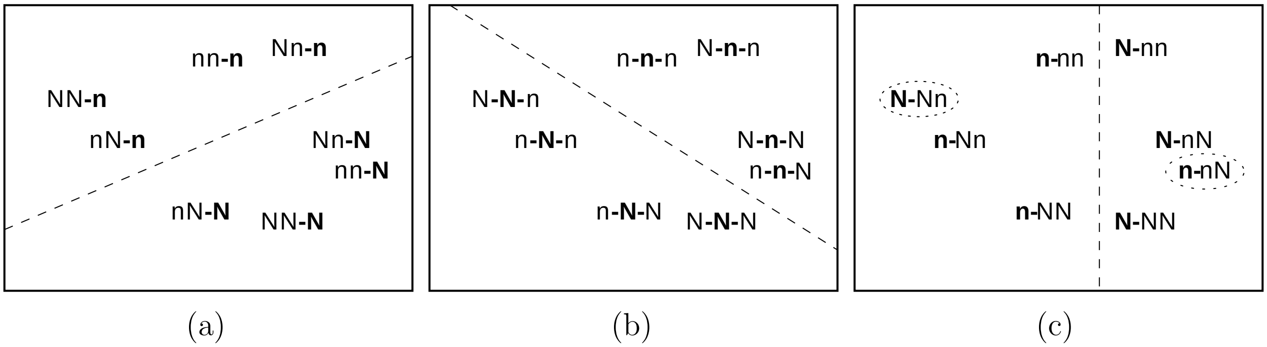

Difference in Comprehensibility
Ratings for German and Dutch
German
Dutch
Difference in Mean Test/Paraphrase Ratings
0.0
0.5
1.0
1.5
2.0
2.5
1 Embedding
2 Embeddings
Sentence Type


Difference in GPE for Center-Embedding
and Cross-Dependency Constructions
Center-Embedding
Cross-Dependency
Difference in Mean Complex/Right-Branching GPE
0.00
0.05
0.10
0.15
0.20
1 Embedding
2 Embeddings
Sentence Type


1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
Mean Activation
Indicates 2VP Preference
Indicates 3VP Preference
Erroneous Activation
Sing Nouns
Plur Nouns
Sing Verbs
Plur Verbs
EOS
Lexical Categories


Comparing Human and SRN
Center-Embedding Data
● —— Human Grammaticality Ratings
□ - - - Grammatical Prediction Error
Mean Grammaticality Ratings
Mean Grammatical Prediction Error
7.0
6.5
6.0
5.5
5.0
0.50
0.45
0.40
0.35
0.30
2VP
3VP
Sentence Type


Comparing Human and SRN Right-Branching Data
● Human Comprehensibility Ratings
□ Grammatical Prediction Error
Mean Comprehensibility Ratings
1
2
3
4
5
6
7
8
9
Mean Grammatical Prediction Error
0.20
0.22
0.24
0.26
0.28
0.30
0.32
0.34
Depth of Recursion
0
1
2
3


nn-**n**
Nn-**n**
NN-**n**
nN-**n**
Nn-**N**
nn-**N**
nN-**N**
NN-**N**
(a)
n-**n**-n
N-**n**-n
N-**N**-n
n-**N**-n
N-**n**-N
n-**n**-N
n-**N**-N
N-**N**-N
(b)
n-nn
N-nn
N-Nn
n-Nn
N-nN
n-nN
n-NN
N-NN
(c)