Learning Lexical Knowledge in Context: Experiments with Recurrent Feed Forward Networks

Steven L. Small

Department of Neurology
University of Pittsburgh

Abstract

Recent work on representation in simple recursive feed forward connectionist networks suggests that a computational device can learn linguistic behaviors without any explicit representation of linguistic knowledge in the form of rules, facts, or procedures. This paper presents an extension of these methods to the study of lexical ambiguity resolution and semantic parsing. Five specific hypotheses are discussed regarding network architectures for lexical ambiguity resolution and the nature of their performance: (1) A simple recurrent feed forward network using back propagation can learn to predict correctly the object of ambiguous verb "take out" in specific contexts; (2) Such a network can likewise predict a pronoun of the correct gender in the appropriate contexts; (3) The effect of specific contextual features increases with their proximity to the ambiguous word or words; (4) The training of hidden recurrent networks for lexical ambiguity resolution improves significantly when the input consists of two words rather than a single word; and (5) The principal components of the hidden units in the trained networks reflect an internal representation of linguistic knowledge. Experimental results supporting these hypotheses are presented, including analysis of network performance and acquired representations. The paper concludes with a discussion of the work in terms of computational neuropsychology, with potential impact on clinical and basic neuroscience.

1. Introduction

Connectionist approaches to the study of language, vision, and memory have led to altered perspectives on the nature of cognition [Churchland and Sejnowski, 1987]. In particular, this work has meant the rethinking of the computer metaphor for the mind, such that human memory does not necessarily have to be a "place" to "store" information and human knowledge can be more than "facts" and "inference rules". These computational concepts can be rejected (or at least questioned) without giving up the metaphor of human mental computation [Feldman, 1989].

Parsing has always played a prominent role in the computational study of human language. One reason for this, of course, was the engineering importance of parsing to early researchers in artificial intelligence; their goal was to access newly devised information resources using "natural" language. Cognitive scientists have also focused on parsing issues, with connectionist approaches contributing increasingly to this attention [Cottrell and Small, 1983; Waltz and Pollack, 1985].

Recent work on linguistic representation in connectionist models [Elman, 1989] has profound significance for the psycholinguistic and computational study of human language. In this work, Elman constructs a feed forward connectionist network [Rumelhart, et al., 1985] with only one (easily implemented) recurrent structure [Cottrell and Fu-Sheng, 1989], and trains it to analyze sentences. For each word presented in a particular sequence, the network must predict the next word expected. When the training has been completed, the network has acquired the ability to perform the desired task. Furthermore, in analyzing (statistically) the nature of the acquired "knowledge", Elman found that similar words, both semantically and syntactically, clustered together. He subsequently used his technique to build a network to learn to predict subject/verb agreements in sentences with relative clauses (differing in number).

This work succeeds for the first time at accomplishing a task that has been attempted a number of times in the recent history of cognitive science [Small and Rieger, 1982] without as much success. This work demonstrates (to a limited, but not insignificant

Full Address: Department of Neurology, University of Pittsburgh, 325 Scaife Hall, Pittsburgh, PA 15261. Phone: (412) 648-9200. Network: small@cadre.dsl.pitt.edu.

degree) that a computational device can learn linguistic behaviors without any explicit representation of linguistic knowledge in the form of rules, facts, procedures, or other symbolic schemes. Furthermore, it does so using a simulation technique that has much closer analogies to the human neurobiological substrate (i.e., neurons and their connections) than do any symbol processing approaches. Of course, there are great differences between connectionist networks (of all kinds) and biological neural "networks", and there are important contributions to cognitive scientific knowledge available from the study of logical representations. However, the ability of a simple network of impoverished computational units to acquire linguistic knowledge is important.

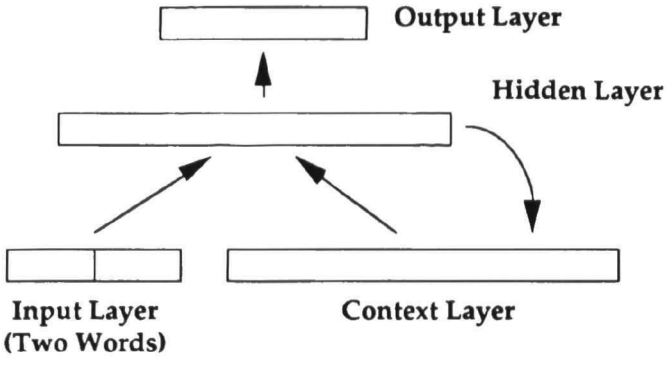

Figure 1: Hidden Recurrent Net with Double Input Buffer

2. Lexical Ambiguity Resolution

Representation of semantic and pragmatic context for lexical disambiguation has been a difficult problem, especially in contexts involving more than one sentence. In the work described here, a simple recurrent feed foward network (as in Figure 1) was trained to predict the next word in a sequence of lexical inputs. The correct desired output word depends on the semantic context of the previous sentence. A set of two sentence "stories" constituted the basic input to the system. Sentence 1a is an example "story" of this type.

(1a) "A man fights. He takes out the assailant."

As part of the experimental method, two simplifying assumptions were made initially, one of which was subsequently lifted. Throughout all of the experiments, the sentences have been simplified by removal of the articles. Sentence 1b shows the example story in the form actually used.

(1b) "Man fights. He takes out assailant."

A further simplification of the experimental method is the merging of "take" and "out" into a single word "take-out". Note that this was done in some but not all of the experiments, and constitutes an interesting part of the experimental design. Sentence 1c shows this input form of the example story.

(1c) "Man fights. He takes-out assailant."

This change simplifies the problem by (a) decreasing the number of total input words and thus the width of the input vector; and (b) decreasing the distance between the contextually important antecedent word and the ultimate ambiguity resolution task word (i.e., predicting the next word after "take out").

The input to the system on any experimental trial included sequences of three or four stories. On each trial, some textual feature or computational parameter was investigated. The two experimental endpoints consisted of the ability of the network to learn the task (i.e., convergence) and the number of trials needed to learn the task. Text comprehension features investigated included the following:

(a) Pronominal gender agreement;
(b) Equivalent contexts for different objects;
(c) Different contexts for the same objects;
(d) Distance between contextual prime and ambiguous word; and
(e) Size of the input buffer.

Computational manipulations were also studied, and aspects that could affect convergence included:

(a) Network learning parameters;
(b) Training instance presentation; and
(c) Network architecture.

The ability of the networks to find solutions to the problems presented suggest a number of things about human linguistic representations, as noted by Elman [1989]. In addition, the ways in which these networks are manipulated to effectuate or improve learning may have implications for language teaching, especially in second language learning or in language learning following damage to the nervous system.

3. Hypotheses and Experiments

All experiments were conducted with feed forward networks and learning by back propagation of error [Rumelhart, et al., 1985].

Five hypotheses motivated the work:

(1) A simple recurrent feed forward network using back propagation can learn to predict correctly the object of ambiguous verb "take out" in specific contexts;

(2) Such a network can likewise predict a pronoun of the correct gender in the appropriate contexts;

(3) The effect of specific contextual features increases with their proximity to the ambiguous word or words;

(4) The training of hidden recurrent networks for lexical ambiguity resolution improves significantly when the input consists of two words rather than a single word; and

(5) The principal components of the hidden units in the trained networks reflect an internal representation of linguistic knowledge.

The experiments were done with networks of three layers, using simple recurrence of hidden units, as shown in Figure 1. The distinct input words were each encoded as if they were orthogonal vectors in an n-dimensional space, where n is the total number of words in the trial (i.e., with six distinct input words, the first one would be encoded as 000001, the second as 000010, and so forth). When the input consists of more than one word (e.g., two words), each word is encoded as before, but with multiple buffer positions, each one consisting of the vector for one word (e.g., the input vector width for inputs of two words becomes 2n). Outputs are encoded separately, with one bit position for each possible output item (i.e., words, concept representations, or case frame data).

The use of one hidden layer (rather than two or three) and the ideal number of hidden units (generally 3 to 4 times the number of coded input units) were empirically determined through many experimental trials. The learning rate (nu or epsilon) was generally kept at 0.6 and the momentum (alpha) at 1.0. Changes in these values had little effect on convergence of the experimental network configurations. No attempt was made to minimize convergence time, and convergence was defined as achieving the correct binary output values for all inputs, with a value < 0.4 defined as zero, a value > 0.6 defined as one, and anything in between undefined, as per the suggestions of Fahlman [1988].

Experiments were conducted with three or four sentence pairs (which we call "stories"), along the lines of those shown above. The training sets presented the input data one word at a time (input buffer size = 1) or two words at a time (input buffer size = 2). Examples of both training instance types for Sentence 1b are shown in Figure 2. Example stories to study both ambiguity resolution and pronoun gender agreement are shown below. These sentences (2a-d) were presented in several different ways during the experiments. The presentations involved either (a) the first three stories or all four stories; (b) a single "take-out" word or two separate words; and (c) either one input word at a time or two input words at a time (as seen in the example training set of Figure 2).

(2a) "Man fights. He takes out assailant."
(2b) "Woman cleans. She takes out garbage."
(2c) "Man loves. He takes out licence."
(2d) "Woman eats. She takes out supper."

INPUT	OUTPUT
man	fights
fights	*
*	he
he	takes
takes	out
out	assailant
assailant	*

Single Word Input Training Set

INPUT BUFFER #1	INPUT BUFFER #2	OUTPUT WORD
man	fights	*
fights	*	he
*	he	takes
he	takes	out
takes	out	assailant
out	assailant	*

Two Word Input Training Set

Figure 2: Example Training Sets with One or Two Input Words

Note that capital letters are not represented, but end of sentence periods are included (as the asterisk in the example training set of Figure 2).

4. Experimental Results

Approximately one hundred experiments were conducted, and some general conclusions are possible on the basis of what was learned empirically from those studies. Fourteen experiments, restricted to three and four story sequences, are summarized in Table 1, and labelled Experiments 1-14. Several parameters that were not varied in these fourteen experiments are not listed in the table, including the number of hidden layers (1), the learning rate (0.6), and the momentum (1.0).

The information included in the table consists of the following: The input buffer width is the number of vectors, each representing a single word, that were input to the network; the networks were presented with either one or two word inputs. The words "take out" were represented in some experiments as a single word "take-out" and in others as two separate words. The input of a "clear signal" (a vector of all zeros) after each epoch aided convergence, as per the empirical observations of Blumenfeld [1989]. The hidden layer fraction is the ratio of hidden layer width to input layer width (before recurrence). A network was considered to converge if it produced the correct results for the training set. This was always true when the mean squared error of the network was less than 0.1. A network was considered to be monotonic when its mean squared error never increased during training. The number of trials shown consists of epochs (complete presentations of the training set).

The hypotheses enumerated above proved to be mostly correct. It was possible to construct feed forward hidden recursive networks to predict the object of the verb "take out" in context (Hypothesis 1). Experiments using the same number of contexts (e.g., "fights") as ambiguous verbs (e.g., "takes out" meaning "knock out with a punch") converged the most readily (these experiments are shown in Table 1). Experiments with more contexts than ambiguous verbs also converged, but not as readily. Experiments with a greater number of ambiguous verbs than distinct contexts did not converge. Experiments including both male and female agents converged more readily than did experiments in which all the stories contained male agents only. The networks were not only able to predict the correct pronoun in context (Hypothesis 2), but actually improved their performance by having this additional nonredundant element of context to use in forming their internal (hidden unit vector) encodings.

By using "take out" as two words in some experiments but as a single word in others, the distance between the contextually relevant antecedent word and the ambiguous word was varied. Better convergence was obtained when it was encoded as one

EXPERIMENT #	1	2	3	4	5	6	7	8	9	10	11	12	13	14
PARAMETERS														
# of stories	3	3	3	3	4	4	4	4	4	4	4	4	4	4
# of "take out" words	1	1	2	2	1	1	1	2	2	2	2	2	2	2
Input buffer width	1	1	1	1	1	1	1	1	1	2	2	2	2	2
Genders represented	M	M	M	M	M	M	M/F	M	M/F	M/F	M/F	M	M	M
# of priming verbs	3	3	3	3	4	4	4	4	4	4	4	4	4	4
# of direct objects	3	3	3	3	4	4	4	4	4	4	4	4	4	4
Clear each epoch?	N	Y	N	Y	N	Y	Y	Y	Y	N	Y	Y	Y	Y
Input width	10	10	11	11	12	12	14	13	15	30	30	26	26	26
Hidden width	40	40	44	44	48	48	56	52	60	45	45	39	65	78
Output width	10	10	11	11	12	12	14	13	15	14	14	12	12	12
Hidden fraction	4	4	4	4	4	4	4	4	4	1.5	1.5	1.5	2.5	3
RESULTS														
Convergence?	Y	Y	N	Y	N	Y	Y	N	Y	Y	Y	N	N	Y
Monotonic?	N	N	N	N	N	Y	Y	N	N	N	N	N	N	N
# of trials (epochs)	887	1294	1160	896	4232	1047	495	3874	611	855	659	2207	2078	1662

Table 1: Summary of Fourteen Prototypical Experiments

word than as two words (Hypothesis 3), though most networks were able to perform adequately when each word was represented separately. The input buffer width had a significant effect on network performance (Hypothesis 4), with two word input experiments converging more consistently and faster than one word input experiments. Hypothesis 5 concerns the nature of the hidden unit vectors following training, and whether or not they constitute a "representation" of linguistic knowledge. This will be addressed in the next section.

5. Network Analyses

Principal components analysis and contribution analysis, a variation suggested by Sanger [1989] specifically for evaluating feed forward networks, were used to analyze the hidden unit vectors. The goal of this analysis was to determine if the hidden unit layer acquired a "representation" of the linguistic knowledge as it learned enough about the structure of the presented stories to predict their outcomes (i.e., the direct object of the context-dependent verbs).

Principal components analysis (PCA) consists of several steps, explained briefly in Fukunaga [1972], aimed at determining a coordinate system for a collection of vectors that maximally separates them, i.e., that organizes them into "components". In a feed forward network with hidden units, these components can be viewed as an encoding of the distributed information acquired by the network in training and used by the network to produce desired outputs for particular inputs. The steps of PCA are as follows:

(1) Compute the hidden unit vector corresponding to each input vector;

(2) Compute the covariance matrix of this array of hidden unit vectors;

(3) Determine the eigenvectors of the covariance matrix; these vectors constitute the new coordinate system;

(4) Sort the eigenvectors by their corresponding eigenvalues;

(5) Translate each hidden unit vector into the new coordinate system.

Contribution analysis simply requires that the hidden unit activations computed in step (1) be adjusted by selected weights between the hidden layer and the output layer of the network. The numerical analysis text by Press et al [1989] includes several of the algorithms required to perform eigenvector computation.

Analyses were performed on the hidden unit layer from Experiment 9 (see Table 1): This experiment involved the collection of the four stories shown in Sentences 2a-d, in which the verb "take out" takes four direct objects, which are primed by the semantic context in a previous sentence, and in which the main actor is either male or female. Traditional principal components analysis was performed, as was a contribution analysis focussing on the distributed hidden unit responsibilities toward particular output words.

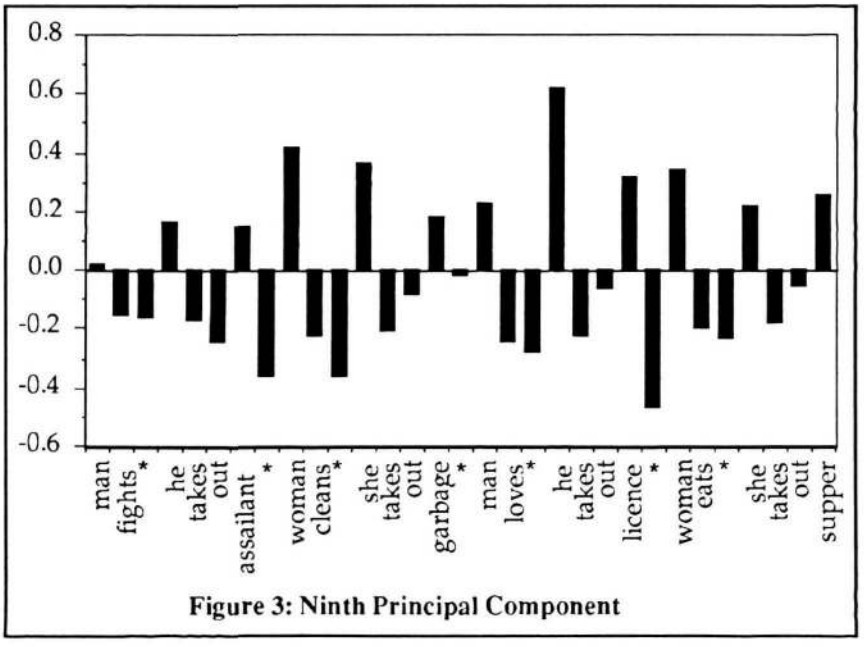

Figure 3: Ninth Principal Component

Three of the most interesting components discovered in this analysis derive from the hidden unit contributions to the output word "garbage", which is one of the primed objects of "take out" that the network learns to predict based on context. These are the components illustrated in the figures. Note that a constant has been added to the raw values to improve the graphic presentation.

Figure 3 consists of a bar

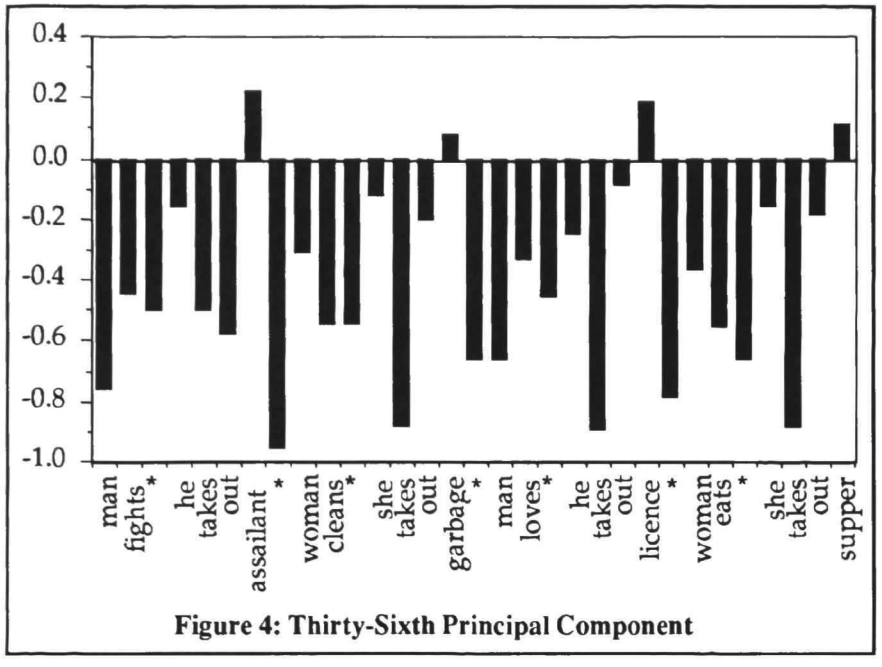

Figure 4: Thirty-Sixth Principal Component

graph illustrating the contribution of the ninth principal component to the decision task of the network. This component discriminates between nouns and other words (i.e., verbs and the periods at the end of sentences). Many of the principal component vectors discriminate among words and word types, and suggest learned linguistic representations (appropriate to the simple linguistic task performed). Other principal components appear not to represent typical linguistic concepts, representing instead heuristically useful information for performing the requested task. In the analyses performed for Experiment 9, for example, one principal component vector seemed to represent a category of pronouns and objects, another represented the words "woman" and all occurrences of the word "take" except the first one, and another the last word of each sentence.

Figure 4 shows that one of the principal components encodes the direct objects of the ambiguous verb "take out". Note that the gender of the sentence subject is subtly represented in these data in the magnitude of the component. The representation of gender contributes to the network's ability to perform the disambiguation task, and can be seen in a number of the principal components, which themselves appear to encode different information. Figure 5 illustrates this point in a principal component that superficially appears to represent the word "out", a word with no semantic role in the experimental stories used here. Significantly, the value of the principal component is maximal when the word "out" occurs in a sentence with a female subject. The network predicts the next word of the input stream by accumulating contextual information; in this case, a readily predictable word that precedes a highly unpredictable one (context excluded) becomes a carrier of contextual information.

As expected, the knowledge gained by these networks, when presented with particular linguistic samples, pertains directly to those samples. The network thus has an inherently heuristic nature; the generalizations (are they representations?) acquired are useful and/or necessary for performing one particular task. Naturally, results based on experiments involving such small corpora of textual samples cannot necessarily be extrapolated to the entirety of human linguistic knowledge and processing.

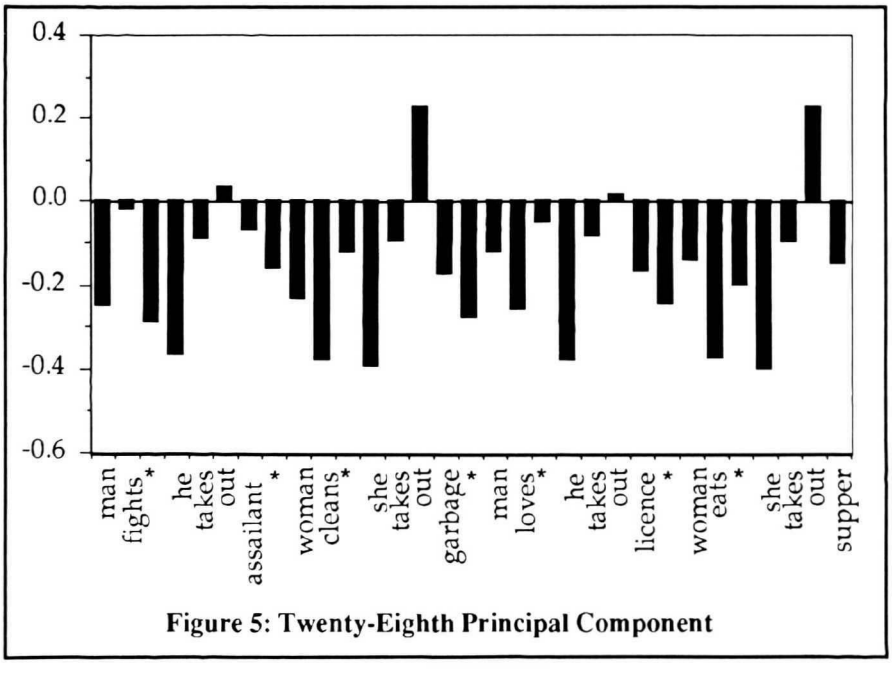

Figure 5: Twenty-Eighth Principal Component

6. Discussion

Much attention has been devoted to the effects of context on human comprehension of sentences and collections of sentences. The relevant context has included local syntactic and semantic features as well as broader elements of textual information. The subject of lexical ambiguity resolution [Small, et al., 1988] has been a productive domain for studies of this type, since understanding the syntactic role and semantics of a word requires knowledge of context at many different levels [Small, 1980].

Linguists have attended to the structural features of sentences and texts that bear on the unambiguous interpretation of subsequent linguistic fragments. Psychologists have employed lexical decision tasks [Tanenhaus, et al., 1979] and auditory evoked potentials [van Petten and Kutas, 1988] to gain information about the temporal sequence of steps performed by the brain to perform word or sentence understanding. While much of this work has been conducted in (presumably) normal users of language, some work has also been done in subjects with language dysfunction, such as Broca's aphasia [Bates and Wulfeck, 1989; Friederici and Kilborn, 1989] or Alzheimer's Disease [Nebes, et al., 1986].

In the current work, a simple recurrent feed forward connectionist network learned to interpret correctly the intended meaning of the words "take out" in context. As noted by Elman [1989], the distributed connectionist approach leads to linguistic performance without explicit rules. Furthermore, the syntactic and semantic structures of language (albeit the very simple examples studied so far) are represented in a distributed non-symbolic form. While in all likelihood, the brain does not employ back propagation learning, it does appear that human learning takes place by weight changes in response to input stimuli (if chemical changes at synapses are viewed as weight changes), and that repetition of stimuli potentiates learning [Lynch, 1986].

7. Conclusions and Future Work

Computational network architectures can learn to perform certain linguistic tasks without any explicitly coded pre-existing linguistic knowledge. In these experiments, simple networks were shown to gain internal linguistic representations sufficient to interpret ambiguous words in context. Furthermore, they were shown to improve performance with (a) shorter distance between contextually important antecedent word and ambiguous word; and (b) increased input buffer size from one word at a time to two words at a time. Both of these processing characteristics have a direct bearing on understanding human performance.

The linear algebraic technique of principal components analysis was used to demonstrate that the network gained a distributed internal representation of various heuristically useful concepts. These concepts include the linguistic notions of "noun" and "direct object", the interesting and useful notion of "the word 'out' in the context of a female agent", and other potentially useful heuristic concepts such as "last word in a sentence" and "period at the end of a contexually important sentence" (i.e., a two word antecedent sentence in one of the simple stories).

Finally, such networks have significant neurological importance. People are subject to a variety of neurological adversities, and the pathophysiology of many are unknown. Computer models of language that can be disrupted to produce deficits analogous to those present in human disease, such as acquired dyslexia [Hinton and Shallice, 1989; Mozer and Behrmann, 1989], may lead to better understanding of these disease processes. In addition to illness, such as stroke and dementia, which produce numerous speaking and understanding (and reading and writing) problems, normal aging also involves changes in linguistic processing. Perhaps a "computational neuropsychology" can shed some light on questions that have been unanswered since Broca [1861].

Acknowledgements

Thanks to the members of the neuroscience community at the University of Pittsburgh who provided helpful comments, advice, and support for the work described here: Audrey Holland, Mark Fitzsimmons, Mac Reinmuth, Gloria Hoffman, and Brad Tanner. Thanks also to Gary Cottrell of UCSD for his help with principal components analysis.

References

Bates, E. and B. Wulfeck, Crosslinguistic Studies of Aphasia, in *The Crosslinguistic Study of Sentence Processing*, MacWhinney and Bates (ed.), Cambridge University Press, Cambridge, 1989.

Blumenfeld, B., A Connectionist Approach to the Recognition of Trends in Time Ordered Medical Parameters, *Symposium on Computer Applications in Medical Care*, Washington, D.C., 1989.

Broca, P. P., Nouvelle Observation d'Aphémie produite par une Lesion de la Partie Postérieure des Deuxième et Troisième Circonvolutions Frontales, *Bullétin de la Societé Anatomique*, 1861, 6: 398-407.

Churchland, P. S. and T. J. Sejnowski, Neural Representation and Neural Computation, Cognitive Neuropsychology Laboratory, The Johns Hopkins University, Technical Report #34, 1987.

Cottrell, G. W. and T. Fu-Sheng, Learning Simple Arithmetic Procedures, *Eleventh Annual Conference of the Cognitive Science Society*, Ann Arbor, 1989.

Cottrell, G. W. and S. L. Small, A Connectionist Scheme for Modelling Word Sense Disambiguation, *Cognition and Brain Theory*, 1983, 6: 89-120.

Elman, J. L., Representation and Structure in Connectionist Models, Center for Research in Language, University of California, San Diego, Technical Report CRL-TR-8903, 1989.

Fahlman, S. E., An Empirical Study of Learning Speed in Back-Propagation Networks, Computer Science Department, Carnegie Mellon University, Technical Report CMU-CS-88-162, 1988.

Feldman, J. A., Neural Representation and Neural Computation, in *Neural Connections, Mental Computation*, Nadel, Cooper, Culicover, and Harnish (ed.), The MIT Press, Cambridge, 1989.

Friederici, A. D. and K. Kilborn, Temporal Constraints on Language Processing: Syntactic Priming in Broca's Aphasia, *Journal of Cognitive Neuroscience*, 1989, 1: 262-272.

Fukunaga, K., *Introduction to Statistical Pattern Recognition*, Academic Press, New York, 1972.

Hinton, G. E. and T. Shallice, Lesioning a Connectionist Network: Investigations of Acquired Dyslexia, Department of Computer Science, University of Toronto, Technical Report CRG-TR-89-3, 1989.

Lynch, G., *Synapses, Circuits, and the Beginnings of Memory*, The MIT Press, Cambridge, 1986.

Mozer, M. C. and M. Behrmann, On the Interaction of Selective Attention and Lexical Knowledge: A Connectionist Account of Neglect Dyslexia, Department of Computer Science, University of Colorado at Boulder, Technical Report CU-CS-441-89, 1989.

Nebes, R. D., F. Boller and A. Holland, Use of Semantic Context by Patients with Alzheimer's Disease, *Psychology and Aging*, 1986, 1: 261-269.

Press, W. H., B. P. Flannery, S. A. Teukolsky and W. T. Vetterling, *Numerical Recipes in Pascal: The Art of Scientific Computing*, Cambridge University Press, Cambridge, 1989.

Rumelhart, D. E., G. E. Hinton and R. J. Williams, Learning Internal Representations by Error Propagation, Institute for Cognitive Science, University of California, San Diego, Technical Report ICS-8506, 1985.

Sanger, D., Contribution Analysis: A Technique for Assigning Responsibilities to Hidden Units in Connectionist Networks, Department of Computer Science, University of Colorado at Boulder, Technical Report CU-CS-435-89, 1989.

Small, S. L., Word Expert Parsing: A Theory of Distributed Word-Based Natural Language Understanding, Ph.D. Thesis, Department of Computer Science, University of Maryland, 1980.

Small, S. L., G. W. Cottrell and M. K. Tanenhaus (ed.), *Lexical Ambiguity Resolution: Perspectives from Psycholinguistics, Neuropsychology, and Artificial Intelligence*, Morgan Kaufmann Publishers, Inc., San Mateo, California, 1988.

Small, S. L. and C. J. Rieger III, *Parsing and Comprehending with Word Experts: A Theory and Its Realization, in Strategies for Natural Language Processing*, Lenhert and Ringle (ed.), Lawrence Erlbaum Associates, Hillsdale, N.J., 1982.

Tanenhaus, M. K., J. M. Leiman and M. S. Seidenberg, Evidence for Multiple Stages in the Processing of Ambiguous Words in Syntactic Contexts, *Journal of Verbal Learning and Verbal Behavior*, 1979, 18: 427-440.

van Petten, C. and M. Kutas, Tracking the Time Course of Meaning Activation, in *Lexical Ambiguity Resolution: Perspectives from Psycholinguistics, Neuropsychology, and Artificial Intelligence*, Small, Cottrell, and Tanenhaus (ed.), Morgan Kaufmann Publishers, Inc., San Mateo, 1988.

Waltz, D. L. and J. B. Pollack, Massively Parallel Parsing: A Strongly Interactive Model of Language Interpretation, *Cognitive Science*, 1985, 9: 51-74.