BEYOND ASSOCIATIVE MEMORY: CONNECTIONISTS MUST SEARCH FOR OTHER COGNITIVE PRIMITIVES

Technical Report AIP - 34

David S. Touretzky

Computer Science Department
Carnegie Mellon University
Pittsburgh, PA. 15232

22 March 1988

This research was supported by the Computer Sciences Division, Office of Naval Research and DARPA under Contract Number N00014-86-K-0678; Reproduction in whole or in part is permitted for purposes of the United States Government. Approved for public release; distribution unlimited.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3 DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; Distribution unlimited
2b. DECLASSIFICATION / DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) AIP - 34	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Carnegie-Mellon University 6b. OFFICE SYMBOL (*If applicable*)	7a. NAME OF MONITORING ORGANIZATION Computer Sciences Division Office of Naval Research
6c. ADDRESS (*City, State, and ZIP Code*) Department of Psychology Pittsburgh, Pennsylvania 15213	7b. ADDRESS (*City, State, and ZIP Code*) 800 N. Quincy Street Arlington, Virginia 22217-5000
8a. NAME OF FUNDING / SPONSORING ORGANIZATION Same as Monitoring Organization 8b. OFFICE SYMBOL (*If applicable*)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER N00014-86-K-0678
8c. ADDRESS (*City, State, and ZIP Code*)	10 SOURCE OF FUNDING NUMBERS p4000ub201/7-4-86 PROGRAM ELEMENT NO: N/A; PROJECT NO.: N/A; TASK NO.: N/A; WORK UNIT ACCESSION NO: N/A

11. TITLE (*Include Security Classification*) Unclassified

Beyond Associative Memory: Connectionists Must Search for Other Cognitive Primitives

12. PERSONAL AUTHOR(S) Touretzky, David S.

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (*Year, Month, Day*)	15. PAGE COUNT
Technical	FROM 86Sept15 TO 91Sept14	1988 March 22	6

16. SUPPLEMENTARY NOTATION

17 COSATI CODES			18 SUBJECT TERMS (*Continue on reverse if necessary and identify by block number*)
FIELD	GROUP	SUB-GROUP	1. connectionism 2. associative memory 3. natural language

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

See reverse side.

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

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

SECURITY CLASSIFICATION OF THIS PAGE

Abstract

Many recent connectionist models can be categorized as associative memories or pattern classifiers. Viewed at the right level of abstraction, the two are the same. Connectionists sometimes appear to be trying to squeeze all of cognition into the associative memory paradigm, perhaps because it's the only thing they know how to implement with gradient descent learning algorithms. But the combinatorial structure of thought and language indicates that the answer to "How can slow components think so fast" lies beyond mere associative recall. We must search for additional cognitive primitives that can be implemented in parallel hardware. One modest successor to associative recall is considered here.

Proceedings of the 1988 AAAI Spring Symposium Series. Parallel Models of Intelligence: How Can Slow Components Think So Fast?

Beyond Associative Memory: Connectionists Must Search for Other Cognitive Primitives

David S. Touretzky

Computer Science Department
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

Abstract

Many recent connectionist models can be categorized as associative memories or pattern classifiers. Viewed at the right level of abstraction, the two are the same. Connectionists sometimes appear to be trying to squeeze all of cognition into the associative memory paradigm, perhaps because it's the only thing they know how to implement with gradient descent learning algorithms. But the combinatorial structure of thought and language indicates that the answer to "How can slow components think so fast" lies beyond mere associative recall. We must search for additional cognitive primitives that can be implemented in parallel hardware. One modest successor to associative recall is considered here.

1 Associative Memory and Pattern Classification

If an associative memory is trained on a set of patterns $\{P_i\}$ and then exposed to a novel pattern P^*, it will usually map the new pattern to the closest P_i. In matrix autoassociators such as Hopfield nets or Anderson's "brain state in a box" model, any subset of the pattern can serve as the cue for retrieving the whole. These models contain recurrent connections, so their behavior is described by differential equations and they typically require several iterations to settle into a stable state. Their memories are fixedpoints of a dynamical system.

A second family of associative memory models permits only feed-forward connections; the weights are learned by backward error propagation or the perceptron learning rule. Feedforward models can acquire much wider sets of memories than matrix models, by using intermediate or "hidden" units to recode the input as necessary [12]. A hetero-associator which maps an input pattern I_i to an output pattern O_i can be viewed as an auto-associator in which $P_i = \langle I_i . O_i \rangle$ and $P^* = \langle I^* . 0 \rangle$.

A third family of associative models, which includes competitive learning and adaptive resonance networks, maintains a prototype for each learned class. A prototype is represented by the connections made to a dedicated unit. Novel inputs are associated with the prototype whose unit they excite most strongly. Learning new prototypes requires recruiting new units.

Associative architectures have been applied to three broad classes of problems: pattern classification, analog function approximation, and pattern transformation. When used for pattern classification, each class of patterns is named by the single O_i to which all I^* in the class are mapped. When used for function approximation, the goal is to produce an O^* close to $f(I^*)$ after training on a set of pairs $\langle I_i, f(I_i)\rangle$. In pattern transformation problems the output pattern contains parts of the input rearranged or transformed in some discrete, systematic way. The early success of associative memories in the first two areas has obscured some important facts about their limitations in the third.

Perhaps the best-known connectionist pattern classifier is Sejnowski and Rosenberg's NETtalk program [14]. NETtalk maps a seven letter window of English text into a 26 bit output pattern that determines the pronunciation of the window's middle letter in that context. The output consists of 23 articulatory features plus three bits for stress and syllable boundary information. By scanning the window across a body of text and using the back propagation learning algorithm to train the weights, NETtalk can be taught to "read aloud." What NETtalk actually learns is a set of 26 binary decision problems, which it solves in parallel. However, since the 26 output units share the same set of hidden units, and certain correlations exist among the outputs (e.g., the articulatory features LABIAL and VELAR are mutually exclusive), the 26 classification problems aren't learned independently.

Lapedes and Farber have shown that back propagation can learn to predict the behavior of very complex (chaotic) functions with better accuracy than previous numerical techniques [7]. Function approximation by back propagation is especially promising for control problems, such as predicting the dynamics of a robot arm from a collection of actual trajectories.

One of the interesting properties of back propagation is that the hidden units evolve to detect regularities in the input space. Hinton's work on learning the structure of family trees offers a particularly striking example of this effect [5]. But the regularities the network discovers are not accessible to introspection; the network cannot reason about what it has learned. For example, the network has no way to compare the regularities discovered in one group of input units with those discovered in other groups.

2 Pattern Transformation by Multilayer Perceptrons

A pattern transformation machine is potentially computationally universal, since any function may be described as a transformation from inputs to outputs. But the class of transformations that are learnable in reasonable time by direct induction from training data is limited.

Allen reports experiments in using back propagation to translate English sentences into Spanish [1]. Starting with a context-free subset of English with 3300 sentences in it, when the network was trained on 99% of them it showed good performance on the remaining 1% (33 sentences). This demonstrates that certain kinds of complex syntactic operations can be learned by back propagation – but only if one is willing to pay the price of near-exhaustive enumeration.

One of the best-known pattern transformation machines is the Rumelhart and McClelland verb learning model [13]. This model takes as input a phonetic representation of a present tense English verb and

produces as output the phonetic representation of the past tense. The model has a single layer of trainable weights. After training on a mixture of regular and irregular verbs, the model is able to predict the past tense of novel verbs by transforming their phonetic representations in accordance with the rules that govern past tense formation in English. It induced these rules from the training data.

Another very interesting pattern transformation machine is the case role assignment model of McClelland and Kawamoto [8], which was able to correctly assign the roles agent, patient, with-PP-modifier, and instrument to noun phrases in novel sentences while simultaneously performing lexical disambiguation. As in the verb learning model and the NETtalk model, what this feedforward network was really doing was solving a collection of binary pattern classification problems in synchrony.

3 Why Associative Memory Is Insufficient

The triumph of associative models is that they provide evidence in support of the connectionist hypothesis that problems that appear to require sequential, symbolic inference may be amenable to parallel solution by some sort of continuous dynamical system [15]. But the evidence is only suggestive. No one should believe that human-like cognitive mechanisms are efficiently constructible by back propagation. Rule-following behavior can be successfully approximated by back propagation networks only when the behavior is relatively simple (no combinatorial structure), or the training data covers most of the input space, as in Allen's language translation experiment. In other words, the brute force associative approach to intelligence doesn't scale.

In focusing on simple, trainable pattern recognition machines, connectionists appear to be ducking the hard problems in cognition, as critics of the approach have been quick to point out. These problems include:

- The compositional structure of language [4]. People's ability to correctly interpret novel sentences composed of familiar words, and to put words together in novel ways, cannot be reduced to learning correlations between input and output patterns in a back propagation network – unless the training set is near-exhaustive.

- The need to express complex relations between objects, as in Drew McDermott's "she is more outgoing with her fellow graduate students than with me, her advisor" [9]. The ability to relate concepts not previously juxtaposed is part of what distinguishes inference from mere associative recall.

- The need for variables and variable binding [11]. Constraints imposed by certain uses of variable binding may well be the primary serializing constraint on cognition [10], but variable binding is nonetheless essential for retrieving and applying knowledge. Variable binding cannot be replaced by associative memory unless one trains on virtually all the variable values in advance.

There have been some attempts at addressing these problems by designing specialized network architectures. These include Touretzky and Hinton's distributed connectionist production system, DCPS [18];

Touretzky's BoltzCONS, an architecture for manipulating linked-list structures [16]; Derthick's micro-KLONE: a connectionist version of the KL-ONE family of knowledge representation languages [2]; Dyer and Dolan's connectionist model of role instantiation in scripts [3]; and Touretzky and Geva's DUCS, a connectionist frame system [17]. These models did not evolve by subjecting a network with random initial weights to voluminous amounts of training data. They were designed methodically, using such techniques as coarse coding, lateral inhibition, and simulated annealing search to produce particular sorts of symbol processing behavior.

4 Another Cognitive Primitive

In this last section of the paper I will focus on a simple cognitive primitive, "appropriate substitution," that highlights the failure of the simple associative approach, but I believe is ripe for connectionist implementation. Consider the linguistic phenomenon known as metonymy, in which one concept plays the part of another. An example taken from Lakoff [6] is the waitress' observation

(1) The ham sandwich just spilled beer all over himself.

Here the sandwich is standing in for the customer. Metonymy is quite common in language. Consider:

(2) John cut an apple from the tree.

What John actually cut was neither the apple nor the tree, but rather the stem that connected the apple to the tree. To correctly understand (2) one first needs a way of detecting that the unique selectional restrictions created by the combination of "cut" and "from the tree" are not met by the direct object "apple." This motivates a search for something related to "apple" – in an appropriate way – that could substitute for it as the patient in the cut act. The problem is complicated by the fact that the meanings of the other words in the sentence aren't fixed. In particular, "cut" is a polysemous verb: its meanings include sever, section, slice, stab, excise, dilute, diminish, traverse, and move quickly.

Comprehending (2) requires the knowledge that apples are *connected* to trees by stems, and one of the senses of cut is to sever a connection. Metonymy clearly involves some sort of associative faculty, but not the simple retrieval described earlier. One does not want to map apple to stem in all contexts. Apple certainly doesn't mean stem in either of these sentences:

(3) John *ate* an apple from the tree.

(4) John cut an apple *beneath* the tree.

The domain of metonymic reference is so rich that one cannot hope to cover the space with a simple associative memory, except by using the entire sentence as context and training on an exponential number of examples. But a hand-designed metonymy machine – one that can dynamically bring together the bits

of knowledge it needs – should be able to interpret truly novel metonymic references without exhaustive training. Even though it might never have considered cutting anything like a mooring rope before, a metonymy machine should have no trouble inferring what is severed in

(5) John cut the boat from the dock.

In my lab at Carnegie Mellon we have begun working on the design of such a metonymy machine.

Acknowledgments

This work was supported by the Office of Naval Research under contract number N00014-86-K-0678, and by National Science Foundation grant EET-8716324. I am grateful to Mark Derthick, Cathy Harris, George Lakoff, Jay McClelland, Dean Pomerleau, and Paul Smolensky for helpful discussions.

References

[1] Allen, R. B. (1987) Several studies on natual language and back-propagation. *Proceedings of the IEEE First Annual International Conference on Neural Networks*, vol. II, 287-298.

[2] Derthick, M. A (1987) Counterfactual reasoning with direct models. *Proceedings of AAAI-87*, 346-351.

[3] Dolan, C. P. & Dyer, M. G. (1987) Symbolic schemata, role binding, and the evolution of structure in connectionist memories. *Proceedings of the IEEE First Annual International Conference on Neural Networks*, vol. II, 335-341.

[4] Fodor, J. A. & Pylyshyn Z. W. (1987) Connectionism and cognitive architecture: a critical analysis. Manuscript draft.

[5] Hinton, G. E. (1986) Learning distributed represenations of concepts. *Proceedings of the Eighth Annual Conference of the Cognitive Science Society*, pp. 1-12.

[6] Lakoff, G. (1987) *Women, Fire, and Dangerous Things*. University of Chicago Press.

[7] Lapedes, A. and Farber, R. (1987) Nonlinear signal processing using neural networks: prediction and system modelling. To appear.

[8] McClelland, J. L., & Kawamoto, A. (1986) Mechanisms of sentence processing: assigning roles to constituents. In J. L. McClelland and D. E. Rumelhart (eds.), *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, volume 2. Cambridge, MA: MIT Press.

[9] McClelland, J. L. Feldman, J., Bower, G., and McDermott, D. (1986) Connectionist models and cognitive science: goals, directions and implications. Report of a National Science Foundation workshop on connectionism. Available as a technical report from the Carnegie Mellon Psychology Department.

[10] Newell, A. (1980) Harpy, production systems, and human cognition. In R. Cole (ed.), *Perception and Production of Fluent Speech*. Hillsdale, NJ: Erlbaum.

[11] Pinker, S. & Prince, A. (1987) On language and connectionism: analysis of a parallel distributed processing model of language acquisition. Occasional paper #33, MIT Center for Cognitive Science. Revised version to appear in *Cognition*.

[12] Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986) Learning internal representations by error propagation. In D. E. Rumelhart and J. L. McClelland (eds.), *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, volume 1. Cambridge, MA: MIT Press.

[13] Rumelhart, D. E., & McClelland, J. L. (1986) On learning the past tenses of English verbs. In J. L. McClelland and D. E. Rumelhart (eds.), *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, volume 2. Cambridge, MA: MIT Press.

[14] Sejnowski, T. J., & Rosenberg, C. R. (1987) Paralle networks that learn to pronounce English text. *Complex Systems* 1(1):145-168.

[15] Smolensky, P. (in press) On the hypotheses underlying connectionism. *Behavioral and Brain Sciences*, to appear.

[16] Touretzky, D. S. (1986) BoltzCONS: reconciling connectionism with the recursive nature of stacks and trees. *Proceedings of the Eighth Annual Conference of the Cognitive Science Society*, 522-530.

[17] Touretzky, D. S., & Geva, S. (1987) A distributed connectionist representation for concept structures. *Proceedings of the Ninth Annual Conference of the Cognitive Science Society*, 155-164.

[18] Touretzky, D. S. & Hinton, G. E. (in press) A distributed connectionist production system. *Cognitive Science*, to appear.