Possible Implications of Connectionism

Wendy G. Lehnert
Department of Computer and Information Science
University of Massachusetts
Amherst, MA 01003

As far as I can tell the most exciting thing happening in AI these days is the invasion of the brain people (a.k.a. the connectionists). The connectionists haven't really invaded the AI community in the sense of making a planned assault - it just seems that connectionism is the sexiest thing around. The AI community has very suddenly become very interested in connectionist techniques and it is only a slight exaggeration for me to say that all the first year graduate students I meet express an interest in connectionism. So perhaps it would be useful to talk about the status of connectionism with respect to the old formal/commonsense semantic arguments. Let's try to pigeon-hole this new paradigm in terms of our old formal/procedural/episodic/semantic distinctions and see what happens.

1. What About Symbols?

The first thing we have to grapple with is the fact that the connectionists are operating within a sphere of assumptions that is problematic to mainstream AI research. The cornerstone of mainstream AI is the idea of symbol manipulation. Interestingly, many of the most exciting efforts in connectionism (the "Parallel Distributed Processing" (PDP) models described by Rumelhart & McClelland (1986)) do not utilize explicit symbols at all. But this does not prevent PDP systems from manipulating information: it just means that a concept in a PDP system is not present in that system as an explicit data structure. Concepts (and attributes and categories) manifest themselves as patterns of activation distributed throughout a strongly connected network of nodes, where the nodes by themselves signify nothing in particular. Distributed representations of this sort can be manipulated to exhibit useful I/O behavior, but our traditional ideas of data and control fail to provide the descriptive framework needed to understand these systems.

The implications of this are important. In mainstream AI, a successful system can be said to embody a theory of human information processing. But this claim is evaluated on the basis of what we understand about that program. An explanation at the level of machine code is not very useful, but a high level flow chart might be. The PDP systems do not lend themselves to this explanatory aspect of AI very readily.

> "The strength of this more complicated kind of representation does not lie in its notational convenience or its ease of implementation in a conventional computer, but rather in the efficiency with which it makes use of the processing abilities of networks of simple, neuron-like computing elements." (Hinton, McClelland, Rumelhart 1986).

In some sense, the task of understanding how a given PDP system works is very much like trying to understand machine code. This should not be surprising, given the intimacy of PDP models with low-level computing mechanisms, but it does tend to alienate those elements of the AI community who are interested in "understanding" their programs in traditional information processing terms. It is no small accomplishment to stop thinking in terms of primitive symbols, data structures, and procedures, in order to start thinking in terms of input vectors, linear thresholds, and necessary conditions for stabilization.

While the presence or absence of explicit symbols may at first seem to be an insurmountable hurdle to any intelligent comparisons between AI and connectionism, it is sobering to consider what the connectionists have accomplished using distributed representations. Connectionists have traditionally looked at "low-level"" information processing problems: motor feedback, stereoscopic vision processing, visual letter recognition, and lexical access for natural language are typical examples. If the AI community has been slow to embrace the lessons of connectionism, it is because mainstream AI is more concerned with "high-level" information processing: text comprehension, problem solving, scene recognition, and inductive learning are closer to the heart of mainstream AI. But now we are beginning to see connectionism "trickle-up" into higher task orientations.

Connectionist systems are now being designed to:

1. Translate sentences into case-frame representations (McClelland & Kawamoto 1986)
2. Index causal chains for narrative recall (Golden 1986)
3. Handle the script activation problem (Sharkey, Sutcliffe, and Wobcke 1986)
4. Index memory for a case-based reasoner (Stanfill & Waltz 1986)
5. Store and retrieve relational data (Hinton 1986)

These tasks are firmly situated in the realms of "high-level" information processing - or at least they used to be. No one is claiming to have solved these problems, but one cannot resist the feeling that a breath of fresh air is clearing a musty old closet.

2. The TWITIT Methodology

Connectionists are generally attentive to the physical properties and limitations of the human brain. At the same time, they experiment with programmable systems and bend an occasional constraint as needed. They are exploiting an interesting mix of science (brain theory) and engineering (TWeak It Til It Thinks). On the one hand, connectionists are more constrained than traditional AI researchers: AI people do not think in terms of hardware constraints. On the other hand, connectionists have no shame when it comes to actually making something work: The business of finding a correct set of weights (or initial values, or network architecture, or whatever) is closer to the Quest for the Holy Grail than any knowledge engineer has cared to go. The AI community became understandably nervous about the TWITIT paradigm for system design shortly after Samuel's checkers playing system failed to extrapolate up to chess. I suppose we never quite got over that one.

Even so, as far as methodological styles go, the connectionist enterprise seems capable of accommodating both "neats" and "scruffies" (Abelson 1981). The neat AI camp can optimize learning rules, establish tests for Boltzmann-equivalence, and worry about decidability as a problem in linear algebra. While all this is going on, the scruffies can revel in the pursuit of graceful degradation, operate on the basis of elusive concept definitions, and learn from experience. Wherever the chips may fall, it is nevertheless true that the connectionist turf is up for grabs in the mainstream AI community. What is the relationship between formal logic and connectionism? Theories of reminding and connectionism? Opportunistic planning and connectionism? Teams are just now forming and the sides are still being chosen.

3. A ROSE is a ROZE is a ROZ is a WOZ

Having said all that, maybe we can now try to say something about our original topic of discussion: how the connectionists weigh in on the formal/procedural/episodic/semantic scales.

To begin, let's consider the problem of representing word meanings. In traditional AI there are basically two competing approaches to the representation of word meanings. (1) The formalist fans assume a componential view in which a word's meaning is represented by a set of semantic features. (2) The episodic enthusiasts assume a

structuralist position in which the meaning of word must be defined in terms of its relationship to other words and available memory structures. Interestingly, there are PDP models inspired by both viewpoints (Hinton, McClelland, and Rumelhart 1986) describe componential systems, while (McClelland and Kawamoto 1986) discuss structuralist PDP systems.[1]

If we look a bit closer at the PDP models for lexical access, we discover that they are governed by remarkably predictable task orientations. The componential systems are all concerned with the problem of mapping single isolated words to their word senses, while the structuralist systems are all trying to resolve word senses during sentence comprehension. *Plus ca çhange...*

On the surface, at least, it seems that connectionist techniques can be applied to any traditional view one wants to promote. But there are some undercurrents afoot that might tip the balance away from a fully neutral position of non-alignment. The undercurrent to watch is the question of learning.

One of the reasons why connectionists (at least the PDP variety) are preoccupied with learning is because they see no other systematic way to approach the design of large (at least 100,000 nodes) networks which cannot be understood as static data structures. Coincidentally, a similar preoccupation with learning has risen in recent years among the proponents of episodic memory. It is easy to build a limited prototype that illustrates the utility of episodic memory structures - but it is much harder to scale up from that to a practical system which utilizes a lot of episodic knowledge effectively. This parallel is at least suggestive of some common ground, although the lisp-lovers and the TWITIT set will have to stretch considerably in bringing their respective methodologies together. I think it will happen. The episodic camp is populated primarily by closet psychologists, and the TWITIT group seems to be dominated by closet neurologists. Whatever other differences exist, both groups build systems in order to test their theories and this requires a healthy respect for engineering. The engineering components of both groups are sufficiently *simpatico* to encourage a few curious adventurers into crossing over.

The formalists operate with a very different methodological style, one that is dominated by a much more philosophical orientation. The formalists prefer to study knowledge in a competence framework rather than a performance framework. This is the study of knowledge as it might be if we could factor out the imperfections of the hu-

[1] A number of research efforts which qualify as connectionist efforts are not PDP systems since they employ "local" representations rather than "distributed" representations. The work of Small, Cotrell & Small, Waltz & Pollack, and Charniak fall into this category.

man mind that conceives it. Never mind the fact that "Three dogs ate four bones" is problematic only for graduate students – these are the problems we can study without reference to performance criteria or subject data or anything else that relies on a concern for human memory organization. References to "semantic memory" confuse the issue (as do those who take formal semantics seriously as a model of human memory), but the difference in methodological styles is obvious. [2]

The advocates of semantic features, quantification, and intension/intention distinctions, are almost never people who design psychological experiments or worry about models of human information processing as a precursor to intelligent information processing.

Given all this, it seems to me that the formalists will be even more uncomfortable with the TWITIT mentality than they were with the old-style scruffies. Of course there will always be room for people who want to nail down optimal annealing schedules and mathematical foundations. So the job prospects for formalists look healthy if the connectionists stage a complete takeover of AI in the next decade. As for the scruffy AI types, it seems that the future depends on whether one is primarily a closet psychologist or a latent engineer. The engineers will undoubtedly find work in the brave new world (they always do), but the closet psychologists will be interesting to watch. They will either retreat with queasy feelings of paradigm failure, or stage a revolution that's tough to call. If the connectionists should ever come to dominate AI, we will have to deal with the very real possibility that we might be able to simulate something without really understanding it very well at all. But that's another panel discussion altogether.

REFERENCES

Abelson R. (1981). "Constraint, construal, and cognitive science". In the *Proceedings of the Third Annual Cognitive Science Conference*, Berkeley, California. pp. 1-9.

Golden, R. (1986). "Representing causal schemata in connectionist systems." in *Proceedings of the Eighth Annual Conference of the Cognitive Science Society*, Amherst, MA. pp. 13-22.

Hinton, G. (1986). "Learning distributed representations of concepts" in *Proceedings of the Eighth Annual Conference of the Cognitive Science Society*, Amherst, MA. pp. 1-12.

[2]This point was nicely illustrated by Drew McDermott's commentary on Geoff Hinton's invited talk at AAAI-86. McDermott said that whatever else might be nice about connectionism, the connectionists really ought to stop worrying so much about learning. In retrospect, I would have predicted this.

Hinton, G., McClelland, J., & Rumelhart, D. (1986) "Distributed Representations" in *Parallel Distributed Processing: Explorations in the Microstructures of Cognition* - vol. 1 (eds: Rumelhart & McClelland). Bradford Books.

McClelland, J., and Kawamoto, A. (1986) "Mechanisms of Sentence Processing: Assigning Roles to Constituents" in *Parallel Distributed Processing: Explorations in the Microstructures of Cognition* - vol. 2 (eds: Rumelhart & McClelland). Bradford Books.

Rumelhart D.E., and McClelland J.L. (1986). *Parallel Distributed Processing: Explorations in the Microstructures of Cognition.* Bradford Books.

Sharkey N.E., Sutcliffe R.F.E., and Wobcke W.R. (1986). "Mixing binary and continuous connection schemes for knowledge access". in *Proceedings for the Fifth International Conference on Artificial Intelligence.* Philadelphia, PA. pp. 262-266.

Stanfill, C., and Waltz, D. (1986). "Memory-based reasoning." Technical Report No. 86-7. Thinking Machines Corp. Cambridge, MA.