Symbolicism and Connectionism: AI Back at a Join Point

Kevin B. Korb
Dept. of Computer Science
Monash University
Clayton, Vic. 3168, Australia

korb@bruce.cs.monash.edu.au

ph: +61-3-9905-5200
fax: +61-3-9905-5146

Abstract: *Artificial intelligence has always been a controversial field of research, assaulted from without by philosophers disputing its possibility and riven within by divisions of ideology and method. Recently the re-emergence of neural network models of cognition has combined with the criticisms of Hubert Dreyfus to challenge the pre-eminent position of symbolicist ideology and method within artificial intelligence. Although the conceits of symbolicism are well worth exposing, the marriage between connectionism and Dreyfus's philosophical viewpoint is unnatural and much of the disputation between connectionism and "traditional" artificial intelligence misbegotten.*

Keywords: Symbolicism, connectionism, computation, Dreyfus, distributed representation, philosophy of AI.

Area of Interest: Foundations of AI.

1 Introduction

In recent years much has been made of the tensions, disputes and divergent methods of "traditional" artificial intelligence on the one hand — which relies upon the use of purely symbolic, qualitative reasoning, such as that supported by symbolic logic — and connectionism on the other — which prefers to use neural network models of reasoning that have no apparent place for symbols at all. It has commonly been asserted that connectionist AI has opened up new possibilities for investigating mind and reasoning using non-algorithmic, sub-symbolic approaches that would otherwise be unavailable. Connectionist modeling has also received a (partially) sympathetic response from Hubert Dreyfus, who has spent much energy since the 1960s attempting to establish the impossibility of artificial intelligence — at least if using non-connectionist means. Many connectionists have in their turn decided that Dreyfus's arguments offer support in their own disputes with symbolic AI.

Here I wish to sketch out the arguments: that connectionism and other approaches to AI have more in common than has generally been acknowledged; that jingoists on both sides — as they are everywhere wont to do — have greatly exaggerated the differences between them, as well as the relative merits of their own methods; and that Dreyfus's arguments against traditional AI were always at best inconclusive and in any case provide faint support for the 'new' connectionism. The most plausible approach for artificial intelligence is to take full advantage of both neural networks and symbolic inference, perhaps producing "hybrid systems"; indeed, there is no reason for these not to be combined with genetic algorithms and many other techniques ignored by those ideologues on both sides who are repelled by any kind of eclecticism — unlike the eclectic evolutionary forces that have produced our own brains.

2 Traditional AI

Artificial intelligence as a discipline of research began in the 1950s, simultaneously with the availability to researchers of the first electronic, digital computers. This was inevitable inasmuch as interest in the possibility of a mechanized intelligence was aroused by the very first *machines*. In the 1950s all avenues of approach that had any promise of success were pursued. Newell, Shaw and Simon investigated the potential of logical inference in their Logic Theorist, which was magnificently able to reproduce a few dozen of the proofs from Russell and Whitehead's *Principia Mathematica* (Newell et al. 1957). Rudimentary artificial neural networks were investigated, notably in Rosenblatt's studies of the perceptron (Rosenblatt 1958). And automated methods of evolving rules using the model of natural selection, nowadays called genetic algorithms or evolutionary computation, were being developed. (For example, in Selfridge's "Pandemonium," 1959; indeed, Selfridge's paper prominently displays important elements of all three traditions: symbolic rules, genetic algorithms, and layers of nodes tied together via weighted connections. Pandemonium was an early hybrid system.) Curiously, these three strategies for developing artificial intelligence — symbolic methods (logic), neural networks, and genetic algorithms[1] — are just the same three strategies that dominate research today; although only the latter two are considered 'new' approaches, for reasons unclear. (Commonly, their antecedents in the 1950s are either ignored or under-appreciated.)

It is not right, then, to label symbolic approaches to AI traditional, if that is meant to imply that symbolic techniques have temporal precedence. What is closer to the truth is that symbolic methods have *dominated* research agendas, until recently. This is in no small measure due to the very effective hostility of Marvin Minsky to neural network approaches, whose research into their limitations culminated in the book *Perceptrons* (co-authored with Seymour Papert 1969). That pivotal work demonstrated that individual perceptrons could learn to recognize only a severely restricted class of objects (those that are 'linearly separable') and raised doubts about the capabilities of larger networks of neural elements; especially, Minsky and Papert pointed out there the lack of a learning method for networks of perceptrons. This work had a serious constraining influence upon neural network research, and especially upon the funding of neural network research. Nor did genetic algorithms prosper before the 1980s; no doubt their tendency to consume large numbers of processor cycles was partially responsible. Be that as it may, a larger responsibility surely inheres in the theology of symbolicism: during this period those advocating symbolic approaches dominated in the journals, funding committees and classrooms. And they were not shy about rationalizing their position of superiority. The writings of John McCarthy, Alan Newell, Herbert Simon, Patrick Hayes and other leading lights of AI during the 1960s and 1970s collectively seek to establish the idea that symbolic processing in general, and logical inference in particular, is the only *conceivable* road to artificial intelligence.

In this context the objections of Hubert Dreyfus to the whole enterprise constituted a hardly noticeable footnote. They made little headway in the AI community and mostly received dismissive treatment.[2] What has changed in the last decade is that neural networks have become popular again. It's been noticed that the limitations of a single perceptron are not shared by networks; and an effective method of learning in networks, called backpropagation, has been developed. At the

[1] Some would argue that genetic algorithms do not offer an *approach* to AI, being only an optimization technique. I believe the writings of John Holland and others make it clear that there is much more to genetic algorithms than simply function optimization, that it offers a quite different perspective on the nature of machine learning and machine induction.

[2] Terry Winograd was a notable exception.

same time it has been found that symbolic AI programs have run up against a variety of severe difficulties. Expert systems, while often commercially successful, work in large part because they are constrained to deal with very tiny problem domains; as soon as they are asked to perform outside of those domains, their performance collapses. Symbolic machine learning programs, on the other hand, can learn to recognize objects in a class, but only so long as class membership is unambiguous; given noisy data, which is *characteristic* of most real learning problems, symbolic machine learning collapses. And logical inference has no ability to support *defeasible* inference, the drawing of a conclusion that is more doubtful than the premises — what we might call jumping to a conclusion. For example, if we are certain that Tweety is a bird, we will be inclined to believe that Tweety can fly. If *we* draw such a conclusion, we will be in very little trouble should we subsequently discover that Tweety is a penguin; the same cannot be said of a computer program relying primarily upon logical inferences. This last difficulty has spawned an elaborate, but elaborately unproductive, research program for the development of default logics (cf. Korb 1995).

These problems encountered *within* the AI research community have provided much of the impetus to neural net research and simultaneously have raised interest in Dreyfus's complaints about AI. Many, especially many non-AI researchers, believe that Dreyfus has simply won his arguments. If he has, he has done so by default, for the opposition put up by the AI community has been relatively weak. But the lack of an organized opposition is no very fine reason for acceding to Dreyfus's position.

3 Connectionism versus Symbolicism

Before attempting to rebut Dreyfus, it will be worthwhile to consider some of the disputes that have arisen independently between symbolic and connectionist AI. Many of these disputes are misbegotten; many others are conducted exclusively between the most extreme representatives of each camp. One non-dispute, for example, is the idea that symbolic AI is committed to investigating AI strictly via algorithmic programming, whereas connectionist AI is free to wander beyond the realm of algorithms.

What is often considered the definitive statement of the symbolic point of view was presented by Newell and Simon (1976), which advanced the *Physical Symbol System Hypothesis*: that the necessary and sufficient condition for a system to exhibit intelligence is that it be a physical symbol-processing system. This proposal arises fairly naturally from the observation that some Turing machines are universal — they can perform *any* computation that any other Turing machine can perform — and from the Church-Turing thesis that any precise definition of 'computation' will turn out to be equivalent to Turing computability; i.e., universal Turing machines are capable of computing *precisely* the set of all computations, neither more nor less. This last thesis is intuitively well-supported by the fact that all other attempts to precisely define the concept of computation have turned out to be provably equivalent to Turing computability, including those using parallel architectures. Since Turing machines are (abstract characterizations of) physical symbol-processing machines, then if intelligence is thought to arise exclusively because of one's physical computational processes — which is at least a plausible thought — then intelligence must consist of physical symbol-processing.

Further observations lead to *logicism* within AI: formal logic has been fantastically successful as a discipline within the twentieth century; it appears to capture (more accurately, it arose from an attempt to capture) the essence of human inference; inference is at the heart of intelligence. Therefore, the *right kind* of symbol-processing is that supported by logical rules of inference.

This last inference, however, is quite a stretch — it goes well beyond what is clearly supported by its premises. Indeed, I have already pointed out that logicism has failed to deal with any variety of defeasible or inductive inference. In any case, most proponents of symbolic AI have ignored the constraints that would be imposed by a *restriction* to logic programming. Indeed, symbolic AI is not even restricted to the use of *algorithms.* Algorithms are, by definition, procedures with three characteristics: they definitely terminate within finite time; they definitely terminate with the correct answer; they use 'well-understood,' primitive steps (e.g., flipping a coin to choose the next action is not allowed). To pick nits: even a conventional computer program that has a bug in it is not executing an algorithm; bugs are not allowed. But more relevantly, the *heuristic programming* that is characteristic of much of symbolic AI goes beyond algorithmic programming, for heuristics by definition are *not* guaranteed to produce the right answers.

Connectionism is based upon an architecture dramatically different from von Neumann machines, the standard serial architecture for everyday computers. Neural networks are composed of interconnected nodes that are segregated into layers. At the first layer are nodes which respond directly to a vector of input sensors; at the final layer are nodes which take various possible states of activity, providing a vector of outputs; in between are one or more hidden layers. Commonly, each layer of nodes is fully connected with the units in the preceding layer; whether a node is activated (fired) or not depends upon the levels of activation in the preceding layer and upon weights attached to each connection. The weights typically start off being randomly assigned. The neural network learns by adjusting these weights in response to good or bad performance (outputs) via some learning algorithm, such as backpropagation.

Neural networks are clear enough. What *connectionism* may be is not altogether clear. It is some body of belief about how to go about creating an artificial intelligence using neural networks, presumably. Whatever it may be, it should not be the belief that neural networks provide a better approach to AI because they are computationally more powerful. There is a straightforward argument that von Neumann machines and neural networks are computationally equivalent. On the one hand, it is a fairly trivial exercise to wire up a neural network to compute AND gates, OR gates, etc. and latches as well. So it is not particularly difficult to implement a universal Turing machine using a neural network. Therefore, neural networks are at least as powerful as von Neumann machines. On the other hand, neural networks are primarily implemented as software on von Neumann machines. Relatively few neural networks actually consist of *physical* nodes and connections at all. But if a von Neumann machine can emulate a neural network, then trivially a von Neumann machine is at least as powerful as a neural network. Therefore, they are computationally equivalent. (For a more exacting treatment of these computational issues see Franklin and Garzon 1991.)

The only apparent escape from this argument for a connectionist who believed neural networks to be computationally superior would be to emphasize that neural networks implemented as analog machines can take full advantage of the *real-valued* connection weights, whereas a von Neumann emulation, due to limited-precision arithmetic in discrete representations, cannot. But it is hardly clear that an analog machine would in fact behave differently from *all* discrete emulations. To suppose this is to suppose that the analog device is sensitive to the *exact* value of the real-valued weight, *beyond any finite enumeration of its digits whatsoever.*[3]

[3] To the extent that we have evidence that there are chaotic systems in the world we have evidence that there are physical processes which are indeed infinitely sensitive to initial conditions. So the *possibility* exists that analog machines may perform computations importantly different from any digital machine. But it remains thoroughly mysterious, given a chaotic de-

Even supposing someone should be prepared to swallow that, it is not clear that this cuts a difference between connectionism and the rest of the artificial intelligence community. It is true that hard-core symbolicists eschew the use of real-valued quantities. But it is not clear *why* they should do so. In any case many AI researchers — well outside the connectionist fold — do employ, and always have employed, real-valued quantities in their work. All of the many who investigate the use of information theory in inference (e.g., Wallace and Boulton 1968), statistical pattern recognition, Bayesian inference, and evolutionary computation employ real-valued quantities. Although this separates these researchers from hard-core symbolicism, it also shows that the argument from infinite sensitivity to real values fails to demarcate connectionist methods from the rest of AI.

4 Distributed Representation

Much has been made of the fact that neural networks can, and usually do, support distributed representations, whereas symbolic architectures do not. In all probability too much has been made of this. I shall examine here a number of influential claims attributing a strong significance to the distribution of connectionist representations.

It is less than entirely clear what 'distributed representation' is supposed to mean. Van Gelder, in his study of the concept (1991), notes that *extendedness* is not enough to capture what connectionists intend: it is not enough to point out, for example, that when the input vector of a neural network describes a situation in which coffee is present — say, coffee in a mug, or in a cup, or in a glass — there is a common pattern of activation across a number of internal units of the neural network. It is trivial that symbolic architectures either do or can spread their representations across multiple elements as well. The presence of coffee might be represented by 'c', 'o', 'f', 'f', 'e', 'e' in successive locations of memory. Perhaps more pointedly, in a semantic network such as those introduced by Quillian (1968) — a symbolic network with nodes representing objects and events and links relations between them — the presence of coffee might be represented by the pattern of spreading activation from the coffee node across links throughout much of the network, capturing the functional role semantics of the concept of coffee, or what Quillian called its "full concept" (1968, p. 227).

Van Gelder considers the idea of adding to the requirement of physical distribution the requirement that elements participating in a distributed representation be capable of participating in other representations as well. He rejects this (p. 36) for the reason that in symbolic systems variables represent multiple objects as well — over time. However, we can add the further condition that elements in a distributed representation must be able to support multiple representations *at the same time*, since in fact that is a feature of neural network representations. Nevertheless, this again will not be sufficient to rule out symbolic representations: crossword puzzles are symbolic, yet the letters on intersecting squares participate in multiple representations simultaneously. Furthermore, some functional role of a node in a semantic network may partially determine a representational content, and nodes may participate in multiple functional roles simultaneously.

Many commentators contrast distributed representations with the *discrete* representations of symbolic systems. Discrete representations are supposed to be "modular": amenable to being added to or subtracted from a knowledge base independently of other representations. However, if you train a neural net via backpropagation to adopt a new representation, everything in the net will change; so far from these representations being modular, "it simply makes no sense to ask

vice, how to harnass that difference in any meaningful way.

whether or not the representation of a particular proposition plays a causal role in the network's computation" (Ramsey, Stich and Garon 1991, p. 212).

But neither side of this contrast between discreteness and distributedness quite lives up to the demands placed upon it. It is true that advocates of expert systems tout as one of their signal virtues the modularity of their production rules (e.g., Winograd 1975). However, any serious examination of the literature of expert systems will quickly reveal that this is a sham: so far from productions being independent of each other, it is a major worry about either adding or deleting productions from working systems that they frequently thereafter fail at problems previously handled correctly (e.g., Buchanan and Shortliffe 1984, chapter 7). Such changes often force extensive testing and fine-tuning to recover good performance. The idea that the semantics of representations can be well understood without regard for the ways in which those representations causally or computationally interact is simply impoverished.

The claimed *lack* of modularity in neural networks is also dubious. Ramsey et al. (pp. 211-13) describe two neural networks trained by backpropagation which share representations for 16 propositions but not for an additional proposition, which was used in training only the second of them. The connection weights of the two networks are markedly different, which is taken to support the notion that adding or deleting representations without disturbing others is infeasible. What they neglect to note is that it is quite typical for a neural network trained twice on exactly the *same* propositions to adopt radically different connection weights: the weights adopted at the start of training are randomly selected. If the first network were trained on the first 16 propositions and only *subsequently* trained on all 17 propositions (which is a perfectly feasible procedure), it is most unlikely that the before and after connection weights would be nearly so far apart as in the cases Ramsey et al. have chosen to show. Regardless of what they say about the impossibility of observing (or even asking about!) the role distributed representations play in computation, it remains the case that those representations must continue to lead to correct outputs in handling the first 16 propositions even while training on the seventeenth — the backpropagation procedure ensures this. And that will not happen if the connection weights are fluctuating wildly during the second training round. Distributed representations can in fact be added and removed "discretely" — meaning, about as discretely as in many symbolic systems.

The distributedness of connectionist representations has been deployed in arguments that connectionism is committed to a rejection of psychology,[4] by both advocates of connectionism (and therefore "eliminativism" — e.g., Ramsey et al. 1991) and detractors of eliminativism (and therefore connectionism — e.g., Davies 1991). These arguments depend crucially upon the uninterpretability of the distributed representations buried within the neural network. Davies' argument, for instance, is that for cognition to exhibit the kind of systematicity that psychology and linguistics attribute to humans it must employ common states across related inferences (1991, pp. 243f; echoing the critique of Fodor and Pylyshyn 1988). For example, it is not enough to recognize that drinking coffee from a cup will warm you up (in some case) and that drinking coffee from a mug will do likewise, but such recognition must rely upon a common underlying concept of warm coffee. However, in distributed representations there is no common syntax that can be manipulated in such inferences; Davies quotes Smolensky (1988): "These constituent subpatterns representing coffee in varying contexts are activ-

[4] *Not* "folk" psychology. It is a rhetorical move by antagonists of psychology to label what they are rejecting as a matter of folk belief(?). Certainly, the concepts involved have their origins in common sense and common language, but the support for their utility goes well beyond that, as any text in cognitive psychology will demonstrate.

ity vectors that are not identical, but possess a rich structure of commonalities and differences." Hence, according to Davies, "there simply is no strictly common subpattern of activation" (1991, p. 248).

If one emphasizes the *strictness* of commonality Davies is requiring here, the claim will surely end up being true — trivially true. No *two* tokenings of the same symbol are ever *strictly* identical. But as Smolensky was quoted as asserting, there is important (rich) commonality between representations of coffee in different contexts. Indeed, Davies himself confesses that the analysis of neural network representations "may vindicate a higher level of description from whose point of view the approximate and blurred commonalities are just variable realizations of real commonalities" (1991, p. 254). There is no obvious reason why the commonalities available are insufficient to carry the weight of systematic reasoning; indeed, some connectionists have responded to Fodor and Pylyshyn precisely by building neural networks which exhibit forms of systematicity which they had claimed to be impossible (cf. the collection Hinton 1991). If such commonalities are inherently too weak for some reason, it is incumbent upon the critics of connectionism (or advocates of eliminativism) to display that reason.

Perhaps the emphasis was intended to be on the contextual dependencies that connectionist representations show: the activiation pattern for coffee-in-a-cup will likely be characteristically different from (while having much in common with) the activation pattern for coffee-in-a-mug. But a semantic network representation of the same facts will show analogous contextual dependencies, in particular the functional roles of a cup of coffee and a mug of coffee may well be interestingly different.

I do not intend to be read here as claiming that there are no interesting differences between the distributed representations of connectionist networks and the "local" representations of semantic networks, or other varieties of symbolic representation. The differences are quite interesting. The flexibility and resilience of neural network processing is certainly of interest. And the task of interpreting distributed representations is both interesting and difficult. What I am suggesting is that the obscurities in the interpretation of neural networks are greatly magnified by those who would spin them into whole philosophies.

5 Dreyfus

Hubert Dreyfus's objections to artificial intelligence have been many and varied. I can here only treat a few of the more salient ones.

One of Dreyfus's main criticisms of artificial intelligence, central to his *What Computers Can't Do* (1972) as well as his more recent *Mind over Machine* (Dreyfus and Dreyfus 1986), is that the rules typical of artificial intelligence programming are context-free whereas human actions are never performed without regard for context and so cannot be rule-governed. Expert systems technology is largely based upon the firing of rules: determining that a condition is satisfied by (matches) the current state of affairs, and so taking the action specified by the rule. But whether the situation matches a rule is typically determined by a simple-minded pattern-matching algorithm that takes no account of more global aspects of the situation within which the system is operating. In order to have a rule-based system sensitive to the global situation, we would have to be able to represent each possible situation internally within the computer program: we "would have to treat each type of situation the computer could be in as an object with *its* prototypical description" (Dreyfus 1979, p. 52). In effect, we should have to have a separate rule for each possible situation.[5] On the other hand, "human beings, of course, don't have this problem. They are, as Hei-

[5] Curiously, it is just such an inability to cope with situations systematically that Davies attributes to neural networks, suggesting that this renders them inadequate for modeling cognition.

degger puts it, already in a situation, which they constantly revise." Dreyfus goes on to say that human situatedness is not reducible to a set of knowledge representations, for it is based upon moods, current concerns, self-image and in general upon our being *embodied* at a certain place and time in the world. What is supposedly characteristic of computational intelligence is its being *disembodied*, its being strictly portable from one universal machine to another.

This argument, while entertaining, is wrong headed. If intelligence is based upon *physical processes*, as symbolicists have insisted rather than denied, then it is not based upon anything disembodied, for there are no disembodied (immaterial) processes. Intelligent robots have never been thought of as disembodied; if or when they exist, they will be independent players in the social-physical world.[6] Nor is it sensible to claim that the capability of representing some large number of situations, say N, requires the presence of N distinct representations; this assertion reveals a lack of awareness of combinatorics. Trivially, in propositional logic it takes two variables to support the representation of *four* possible situations. Things are no different with other forms of representation. And Dreyfus's notion that moods, concerns, self-image, and spatio-temporal (and cultural!) location cannot be represented within a computer program is backed by nothing more than presumption. *Each* of these has in fact been the subject of investigation by computationalist cognitive scientists.

Another argument Dreyfus pushes against rule-based expert systems (Dreyfus and Dreyfus 1986, chapter 1) is that whereas human novices certainly do use rules — indeed learning and applying rules is just how they begin to function within a new and difficult domain, such as chess — human experts are just those people who no longer need to use rules. Grandmasters can play five-minute chess games "without serious degradation in performance," which certainly cannot be done by following rules! By contrast to expert systems, connectionist programs may well suffice to capture expertise, for neural networks operate without the use of explicit rules (cf. Dreyfus and Dreyfus 1990). (As an aside, it is not true that grandmaster performance does not degrade under time pressure; what is likely true is that non-experts will have a hard time recognizing the relatively small drop-off in performance.)

There are a number of potential confusions in this line of thought. I have already raised doubts that neural networks are properly described as operating without explicit rules at the cognitive level. However, even if we accept Dreyfus's argument at face value, it still would not necessarily follow that artificial intelligence should abandon rule-based programming. The issue hangs in large part upon what you imagine you are doing when you design AI programs. At one extreme one might be attempting to reproduce or simulate intelligence *as it occurs* in humans or other animals. That would be a descriptive approach to AI, what Herbert Simon has called computational psychology (Simon 1983). At another extreme one might be attempting to produce an intelligence *unlike any other*, in a normative AI perhaps. Of course, one can adopt intermediate or mixed strategies of various kinds. My point is this: if humans do not exercise intelligence by exercising rules, this is entirely beside the point for an extreme normative AI, and is also no decisive point against most intermediate flavors of AI. Rules may nonetheless play an important role in developing an artificially intelligent system.

Another issue has to do with how we understand the concept of following rules. When we talk of *our* following some rules, what we ordinarily have in mind is that we *consciously* apply some rules within some endeavor. I

[6] It is undeniable, however, that much AI research has proceeded with scant regard for the issue of embodiment (Lenat's CYC project comes to mind as a prominent example). And I am inclined, with Dreyfus, to consider such approaches too narrow to be very promising as the overarching research programs that they commonly are claimed to be.

take it that it is clear that human experts hardly ever follow rules in this way; it is completely implausible that grandmasters, or anyone else, can play good speed chess in that way. But it does not follow that the cognitive activity of grandmasters is not rule governed. Nobody believes that the planets literally *obey* Kepler's laws of planetary motion, and yet they do obey them (approximately); that is, Kepler's laws jointly form a true (approximative) theory about how the solar system functions. The lack of introspective rule-following by grandmasters is no more telling against the idea that their chess thought is properly modeled using some set of rules than is the lack of introspection by Jupiter and company telling against Kepler's model of the solar system. What processes implement cognition in humans is not answerable via introspection (even though introspection surely provides some relevant evidence; cf. Dennett's discussion of heterophenomenology, 1991).

Dreyfus specifically doubts that there are true theories of cognition in the same sense that there are true theories of the solar system; for he believes that this would commit us to the idea that "everyday practice is ... based on unconscious theory" (Dreyfus and Dreyfus 1990, p. 396). But it is not true that because we are talking about a theory of cognition the cognition has to be using the theory, consciously or unconsciously. All that is required is that the theory describe the causal structure of the cognitive process. It will be sufficient evidence of that if we can use the theory to make appropriate experimental predictions; it is in no way required that when we examine the fine structure of neural interconnections we should find the theory of cognition somehow encoded in there! I am personally not at all persuaded that the kinds of rules that have been deployed in expert systems are sufficient to build up a model of human cognition. But my skepticism is based upon the limitations of that approach, and the richness of alternatives such as neural networks, rather than Dreyfus's curious arguments.

Along similar lines I wish to raise doubts about an interesting objection to computationalism suggested by a reading of Gerald Edelman (1992). Edelman points out that under certain circumstances human and animal learning is accompanied by structural, morphological changes in the interconnections of neurons. For example, violinists have more cortex devoted to sensing and controlling fingers than normal. This variability of neural wiring shows that the brain cannot be hardwired like a computer (p. 27); and the variation induced by learning suggests that learning cannot be thought of as a software process.

But what is software and what is hardware? In computers the boundary is quite fluid: functions that were previously implemented in software may subsequently be built into a VLSI chip. On the other hand, RISC machines achieve a streamlined architecture by expelling complex instructions from the hardware, requiring them to be implemented in software. In general, programmers do not care whether a function is supported directly by hardware or is provided by software implementing a 'virtual machine.' It makes no difference to the programming task (except in speed of execution perhaps). In the brain the boundary is not so much fluid as it is nonexistent, or at least unknown. While it may be seductive to identify neural interconnections in our brains with hardware and synaptic potentiation with software, it is not clear what benefit might derive from such an identification. But the difficulty, or impossibility, of construing the brain as a von Neumann machine defeats neither AI nor the view that cognition is largely a matter of computation. Neural networks perform computations — otherwise they could hardly be computationally equivalent to von Neumann machines. However, they are either *all* software (when being emulated by software running on traditional computers) or they are *empty* of software — as they are typically not *programmed* in any sense, there can be no software to be found there.

6 Conclusion

What is at issue is whether human and animal cognition can be modeled properly via (largely) computational theories. It is not at all at issue in what medium the computations are being performed. Of course, that is an issue for the *biology* of cognition, but not for AI — except to the extent that the medium has an impact on what kinds of computations may be more easily, or more rapidly, performed (granting also that the medium may be important for any *non-computational* aspects of intelligence). The disputes between connectionists and symbolicists have largely been conducted without regard for their underlying similarities and common purpose — and under the pretense that these two paradigms for AI jointly exhaust the possible means for producing an artificial intelligence. Dreyfus's continuing attacks on artificial intelligence can draw solace from connectionism only by distorting both connectionism and AI — for example, by asserting that symbolic AI presupposes what it manifestly does not, that human expertise consists of consciously or unconsciously obeying explicit rules.

That animal cognition is primarily a matter of computation is an empirical hypothesis; it will not be settled by philosophical disputation but by empirical investigation, by the attempt to further develop descriptive AI. Whether normative AI will be successful in constructing an alien, non-animal intelligence is another, distinct empirical question. These matters are very much in doubt — as could hardly be otherwise for such central questions for the youngest of the sciences.[7]

References

Buchanan, B. and E.H. Shortliffe (eds.) (1984) *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project.* Reading, MA: Addison-Wesley.

Davies, M. (1991) 'Concepts, Connectionism, and the Language of Thought,' pp. 229-257 in W. Ramsey, S.P. Stich, D.E. Rumelhart (eds.) *Philosophy and Connectionist Theory.* Hillsdale, New Jersey: Lawrence Erlbaum.

Dennett, D. (1991) *Consciousness Explained.* New York: Little Brown.

Dreyfus, H. (1972) *What Computers Can't Do.* New York: Harper and Row.

Dreyfus, H. (1979) *What Computers Can't Do*, second edition. New York: Harper and Row.

Dreyfus, H. and Dreyfus, S. (1986) *Mind over Machine.* New York: The Free Press.

Dreyfus, H. and Dreyfus, S. (1988) 'Making a Mind Versus Modeling the Brain: Artificial Intelligence Back at a Branchpoint,' *Daedalus* 117: 15-43.

Dreyfus, H. and Dreyfus, S. (1990) 'Towards a Reconciliation of Phenomenology and AI,' in D. Partridge and Y. Wilks (eds.) *The Foundations of Artificial Intelligence: A Sourcebook*, Cambridge: Cambridge University.

Edelman, G. (1992) *Bright Air, Brilliant Fire.* Basic Books.

Fodor, J. and Z. Pylyshyn (1988) 'Connectionism and Cognitive Architecture: A Critical Analysis,' pp. 3-71 in S. Pinker and J. Mehler (eds) *Connections and Symbols.* Cambridge, MA: MIT.

Franklin, S., and Garzon, M. (1991) 'Neural Computability,' *Progress in Neural Networks* 1: 127-145.

Hinton, G. (1991) (ed.) *Connectionist Symbol Processing.* Cambridge, MA: MIT.

Korb, K. (1995) 'Inductive Learning and Defeasible Inference,' *The Journal of Experimental and Theoretical AI 7*, 291-324.

Minsky, M. and Papert, S. (1969) *Perceptrons.* Cambridge, MA: MIT.

Newell, A. Shaw, J.C. and Simon, H. (1957) 'Empirical Explorations with the Logic Theory Machine: A Case Study in Heuristics,' RAND Corp. Report P-951. Reprinted in E.A. Feigenbaum and J. Feldman (eds.)

[7] I wish to thank Jack Copeland for his helpful commentary and Binh Pham for her kind provision of facilities.

(1963) *Computers and Thought*, New York: McGraw-Hill.

Newell, A. and Simon, H. (1976) 'Computer Science as Empirical Inquiry: Symbols and Search,' *Communications of the ACM* 19: 113-126.

Quillian, R. (1968) 'Semantic Memory,' pp. 227-270 in M. Minsky (ed.) *Semantic Information Processing.* Cambridge, MA: MIT.

Ramsey, W., S.P. Stich and J. Garon (1991) 'Connectionism, Eliminativism, and the Future of Folk Psychology,' pp. 199-228 in W. Ramsey, S.P. Stich, D.E. Rumelhart (eds.) *Philosophy and Connectionist Theory.* Hillsdale, New Jersey: Lawrence Erlbaum.

Rosenblatt, F. (1958) 'The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain,' *Psychological Review* 65: 386-408.

Selfridge, O. (1958) 'Pandemonium: A Paradigm for Learning,' *Symposium on the Mechanization of Thought.* London: HM Stationery Office.

Simon, H. (1983) 'Why Should Machines Learn?' pp. 25-37 in R.S. Michalski, J.G. Carbonell and T.M. Mitchell (eds.) *Machine Learning*, Los Altos, Calif: Morgan Kaufmann.

Smolensky, P. (1988) 'On the Proper Treatment of Connectionism,' *Behavioral and Brain Sciences 11*, 1-74.

van Gelder, T. (1991) 'What is the "D" in "PDP"? A Survey of the Concept of Distribution,' pp. 33-59 in W. Ramsey, S.P. Stich, D.E. Rumelhart (eds.) *Philosophy and Connectionist Theory.* Hillsdale, New Jersey: Lawrence Erlbaum.

Wallace, C.S. and Boulton, B.H. (1968) 'An information measure for classification,' *The Computer Journal, 11*, 185-194.

Winograd, T. (1975) 'Frame Representations and the Procedural/Declarative Controversy,' pp. 185-210 in D.G. Bobrow and A. Collins (eds.) *Representation and Understanding.* New York: Academic Press.