Appears in *Informatica* 19(4):501–511, Nov. 1995 (special issue *Mind <> Computer*).

Stuffing Mind into Computer: Knowledge and Learning for Intelligent Systems

Kevin J. Cherkauer
Department of Computer Sciences
University of Wisconsin-Madison
1210 West Dayton St., Madison, WI 53706, USA
Phone: 1-608-262-6613, Fax: 1-608-262-9777
E-mail: cherkauer@cs.wisc.edu
http://www.cs.wisc.edu/~cherkaue/cherkauer.html

Keywords: artificial intelligence, knowledge acquisition, knowledge representation, knowledge refinement, machine learning, psychological plausibility, philosophies of mind, research directions

Edited by: Marcin Paprzycki

Received: May 10, 1995 **Revised:** November 21, 1995 **Accepted:** November 28, 1995

The task of somehow putting mind *into a computer is one that has been pursued by artificial intelligence researchers for decades, and though we are getting closer, we have not caught it yet. Mind is an incredibly complex and poorly understood thing, but we should not let this stop us from continuing to strive toward the goal of intelligent computers. Two issues that are essential to this endeavor are knowledge and learning. These form the basis of human intelligence, and most people believe they are fundamental to achieving similar intelligence in computers. This paper explores issues surrounding knowledge acquisition and learning in intelligent artificial systems in light of both current philosophies of mind and the present state of artificial intelligence research. Its scope ranges from the mundane to the (almost) outlandish, with the goal of stimulating serious thought about where we are, where we would like to go, and how to get there in our attempts to render an intelligence in silicon.*

1 Introduction

The ultimate goal of artificial intelligence (AI) is to somehow implement a very wonderful and complex thing we call "mind" within the confines of an artificial computer. Even if undaunted by the incredible paucity of our own understanding of mind, we may nonetheless find ourselves put off by the sheer complexity and size we usually imagine this machinery must entail. Despite our inability to satisfactorily define intelligence, one component we generally feel must be present is a large store of *knowledge* about every aspect of the world. However, it helps us little to decide, "Let us put everything we know into a computer." How do we represent this knowledge? How do we refine it? And how do we get it into the system? Surely we do not have time to put *everything* in by hand!

Perhaps our systems can use learning to acquire and modify the knowledge they need largely on their own. Instead of trying to stuff our own brains into the computer one bit at a time (Figure 1), perhaps we can write programs that let the computers learn for themselves what they need to know. Learning is, after all, the way humans fill their own brains with knowledge. But how much can we gain from human analogies? Is psychological plausibility a necessity or a curse? Will our machines need emotional motivation in order to be truly successful learners? The questions, as always, come thick and fast.

In this paper we will take a moment to examine these issues of knowledge and learning in the light of both current philosophies of mind and the present state of artificial intelligence research. It is not often, in the world of technical papers, that we allow our thought processes to roam free. That

Figure 1:

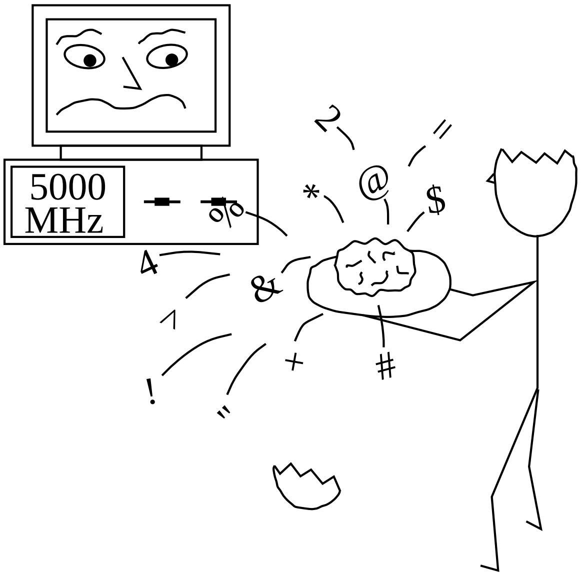
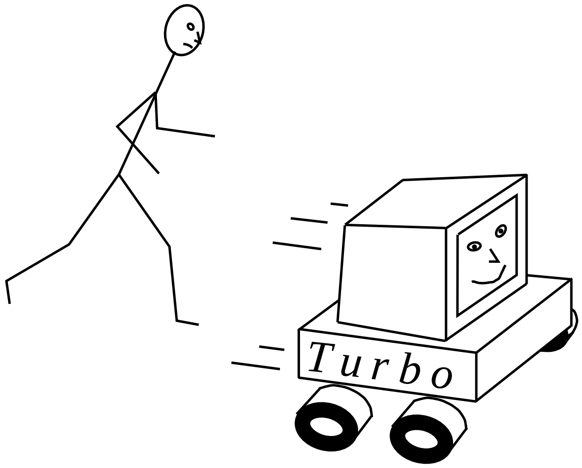


Figure 2:



is the main goal of this paper—to visit some of the wild pastures of imagination that spawned the field of AI in the first place. We hear these days that all those far-flung dreams of intelligent computers from decades ago are still as out of reach as ever. We spend too much of our time being apologetic, trying to present AI advances in as narrow a scope as possible, almost as if we wish them to appear insignificant in order to avoid accusations of chasing hopeless fantasies. It is indeed important to keep a firm grip on reality—I do not think anyone would argue otherwise. But if we are truly to achieve wonders, we must first allow ourselves to imagine them. I hope you will join me in doing so!

2 How Should Our Systems Acquire Knowledge?

The question of how to get knowledge into our systems is a key issue in building intelligences. Most expert systems currently acquire knowledge through painstaking hand programming by a knowledge engineer working closely with a domain expert. A major goal of AI is to produce machines that perform intelligent tasks, so a dedicated AI researcher may suggest that the best answer to our question, "How should our systems acquire knowledge?" is, "Why, through machine learning, of course!" Some obvious advantages of automating the knowledge acquisition process through machine learning (ML) are speed and accuracy of rule construction. However, to succeed in this endeavor we must somehow develop ML techniques which are as good at creating sets of rules for specific domains as an expert human knowledge engineer. This just pushes the problem of emulating expert behavior one level deeper: in trying to avoid hand coding a program that embodies the knowledge of a domain expert, we find we must now hand code a program that embodies that of a knowledge engineer!

We may still manage to tackle this problem if we can find some way to make the knowledge engineer's knowledge easier to program than the domain expert's. Humans use their knowledge and intelligence to construct expert system knowledge bases. Our comparatively dim-witted computers' only chance to overcome their own lack of insight is their blinding speed and tireless persistence (Figure 2) and their utter disdain for the human propensities toward fatigue, boredom, distraction, careless mistakes, and other such egregious vices. Since these are the computer's fortes, we must exploit them.

For instance, we can have our machines search very large numbers of possible rules and rule fragments to find a good set. Whereas a human knowledge engineer examines only a few alternative rules, banking on the domain expert's deep understanding of the problem to insure a good solution, the less knowledgeable computer must succeed through perseverance. The FOIL system [38] is one example of a large-scale search approach to construct predicate calculus rules describing a domain. Part of my own recent work [8, 9] has concentrated on high-speed parallel search methods to sift through hundreds of thousands of potentially useful features for representations that make learning easier.

Computers have an advantage over people in dealing with huge volumes of data. In many cases a problem is too complex and poorly understood for people to construct effective rules to solve it. All that is really available is a large set of raw data. Object recognition, image understanding, speech production, argument construction, complex motor skills, breast cancer prognosis, and protein folding prediction are all real-world problems that fit this description. Some of these are problems of perception and action that humans accomplish effortlessly, yet we cannot articulate how we do so. Others are more abstract problems of interest to science and medicine. All of them have been the subject of machine learning research (e.g. [7, 25, 35, 37, 40, 44, 45, 46]).

This is not to say we should require our machines to learn absolutely everything from scratch. We should certainly take advantage of existing domain knowledge, both low- and high-level, to the extent we can afford it. There is no reason to learn logical inference rules from first principles when we can easily code them into a knowledge base. Likewise, if a domain expert can provide partial sets of high-level rules or other advice, this will jump-start the system and reduce the amount learning time and data required [21, 34, 50]. Guidance from domain knowledge may also be crucial to prevent so-called "over-searching" [39], or the discovery of spurious correlations during learning. Unfortunately, human expertise is too expensive to allow us to hand code everything in a system of the size and complexity needed for intelligence. The builders of the monumental Cyc knowledge system, though willing to invest large amounts of effort to hand code much of the knowledge, nonetheless advocated automating this process as much as possible through ML techniques even from the early stages [19], and they continued to add learning mechanisms over the years [17]. As the intelligent systems we design become increasingly sophisticated, we have no choice but to adopt machine learning techniques as facilitators. To reach human-level intelligence, an artificial system must be enormously more complex than anything we have created to date. The journey to machine intelligence will be shortest if we continue to develop and apply the powers of machine learning on this quest.

3 What Form Should the Knowledge Take?

A serious problem with using ML for knowledge acquisition is what Michalski terms the "knowledge ratification bottleneck" [23]. That is, for applications in which malfunction could have costly, critical, or even life-threatening consequences, any knowledge a system uses must be closely examined for correctness. It is difficult enough to do this with large knowledge bases written by humans; the problem is only compounded if they are cobbled together automatically by a machine. Michalski contends that in such situations, the explanation capabilities of ML systems must be well developed, and the knowledge representation used should be comprehensible to humans. These constraints seem to favor the symbolic, rule-like representations we have spoken of so far over other alternatives like connectionism.

Or do they? Are huge rule bases of the scale needed to simulate human-level intelligence any more comprehensible than artificial neural networks (ANNs)? On the other hand, why can not connectionist representations be made as understandable as rules? Mitchell and Thrun [25] develop ANNs which model various primitive robot actions and then treat these networks as if they were rules. Others have developed methods that allow the extraction of symbolic rules from trained neural networks [10, 13, 14, 42, 48, 49], so the two representation styles are not as irreconcilable as they look.

The question of what form knowledge should be *stored* in relates to the question discussed in the previous section of how a system *acquires* knowledge. If learning is used to do this, many different internal representations are possible, rules and ANNs among them. The hand coding approach, in which humans construct the knowledge base, generally favors a symbolic storage representation. However, there exist machine learning systems that can store and refine initially symbolic knowledge in connectionist ANNs (see Section 5), so there is no reason hand-coded knowledge must remain in its original form.

There are arguments other than understandability for preferring symbolic knowledge structures. Higher-level human cognitive processes operate in an apparently symbolic fashion, perhaps suggesting we should use similar approaches in computers. However, a connectionist might reply that the perceived symbolic nature of our reasoning processes is an illusion, as the brain is a connectionist device. A third person might dismiss both of these arguments, claiming it does not matter how *humans* solve problems if our goal is to build *machines* to do the same. The classic conflicts over psychological and physiological plausibility persist. Let us explore these conflicts further in the next section.

4 Psychological Plausibility: Friend or Foe?

A common argument against using rules to describe knowledge is that of psychological (and sometimes physiological) plausibility. The brain is physically a connectionist device. It is tempting thus to equate psychological plausibility with connectionist implementations, but in fact it is less clear how much the details of abstract cognition depend directly on the connectionist nature of the hardware. It is not unknown for discussions of these questions to become quite animated, especially as there seem to be almost as many points of view as there are interested parties. An imaginary conversation may help us to better understand the extent of the rifts that exist.[1]

Engineer: Psychological plausibility is just a meaningless hoop to jump through, completely superfluous to our goal of building thinking machines! It's hard enough to get anything like intelligence out of a computer even without a bunch of arbitrary anthropocentric constraints. Now you're telling me you won't be satisfied with mere human-*like* intelligence, but you insist on human-*structured* intelligence to boot! Next you'll demand android bodies, vat-grown neural brains, and probably even—*emotions!* We should just go with what works regardless of what it looks like.

Psychologist: How can you take such a position when the human mind is our only example of advanced intelligence? Only incredible arrogance would let us imagine we can start from scratch, ignoring everything psychology has to tell us, and do a better job. If we ever want our systems to speak to us as peers, they will have to understand things the same way we do. It is sheer folly to attempt a computer intelligence that conflicts with our accumulated body of psychological knowledge.

Neurobiologist: *(Clapping hands.) Bravo!* But the psychologist does not go far enough. I'll grant that we know a few things about human cognition, but we have even more specific knowledge about the hardware that implements it. We know exactly how neurons fire, what chemicals they use to transmit signals across synapses—even their patterns of connection in some parts of the brain. AI's best bet is to simulate this hardware as closely as possible, as it is the only thing we thing we have a concrete description of.

Engineer: Ah ha! *(Dons a smug look.)* I *knew* someone would want vat-grown brains!

Philosopher: *(With a sly look.)* Hold on! Why are we limiting our vision to puny, human-like machine intelligences? Shouldn't our goal be to create machines that are *smarter* than people? We can't copy knowledge from adults to babies or put people through a thousand years of education, but we can build computer memories big enough to hold entire libraries and processors fast enough to digest them. Does piscine plausibility help us build nuclear submarines? Does avian plausibility help us build airliners? *(Throws up hands.)* Absolutely not! In fact, these things merely hold us back!

Indeed, there seem to be two diametrically opposed and largely antagonistic camps with respect to this issue: those who believe that psychological (or even biological) plausibility is essential to producing an intelligent artificial system, and those who believe these requirements are merely contrived obstacles that slow our progress or limit the

[1] Of course, there are many more points of view within a given field than these caricatures present.

goals we set for AI. One is tempted to say that what we need most of all is a moderate voice, a compromiser, a fence-sitter—perhaps even a

Politician: Ah, you people are hopeless. The problem is hard enough without all these religious schisms. We should use what ideas we can from psychology without promising to produce a psychologically plausible computer system. We should look to neurobiology for insight without promising vat-grown brains—or even neural networks. We should apply machine learning without promising that every component of the final system will be automatically generated instead of hand programmed. We should follow visions from philosophy without promising to realize them without revision (if I may be so bold as to pun). In short *(waving hands)*, we should take everything we can get our hands on and guarantee nothing in return! (Er...that didn't come out quite right....)

Underlying all this waffling is an important issue which has so far remained implicit, and that is the distinction between the hardware on which an algorithm is implemented and the algorithm itself. Von Neumann [27] states unequivocally that, while we understand the abstract concepts of logic and mathematics in a symbolic way, these concepts must necessarily be implemented very differently in human brains than in digital computers because of fundamental hardware differences. The brain is a massively parallel, low-precision device that encodes information robustly via statistical patterns and performs relatively short chains of calculation. Digital computers are (much more) serial and depend on long chains of brittle, high-precision calculations in which a single corrupted bit can cause a system crash. When we speak of logic and mathematics, we are really using a pseudocode that describes the algorithm without saying anything about the details of implementation. Symbolic descriptions of high-level natural languages and reasoning systems tell us little about their biological implementations. The implication is that they will tell us no more about how to implement them in digital computers.

For these reasons, I believe the most fruitful approach to resolving the controversy of this section is to view psychology and biology as tools for discovering the *algorithms* the human brain runs. Knowing the algorithms, we can then focus on producing the (radically different) *implementations* required for digital computers where this appears suitable. We must keep in mind that the brain's massively parallel algorithms may often be impractical under serial reimplementation due to time or space requirements [43]. There will also be many cases where psychological and biological study are unable to glean the specific algorithm the brain uses to solve a given problem. In these situations, we must resort to more bottom-up engineering that takes best advantage of the strengths of digital computers to arrive at alternate solutions. One example of success using this approach is that of chess playing programs. Although few would argue that human grand masters and computers implement the same chess playing algorithms, it is impossible to deny that computers can play chess at the grand master level. In this problem, an alternate algorithm based on high-speed serial search has achieved the same quality of results as the very different process of high-level human reasoning.

To summarize, psychology and biology should be treated as two tools among many the AI researcher can use to gain insight into methods of intelligent problem solving, but they should not be seen as the only legitimate tools in the arsenal. While the computational properties of the brain and digital computers do overlap, they are far from identical. We can gain algorithmic insight from the brain's solutions, but we will certainly need to tailor these solutions, and often radically alter them, to fit the differing properties of the computer. I do not think there is much to gain by demanding psychological plausibility, whatever that may be, in computer systems that are by nature so unlike the brain, nor do I think there is any real justification in this context to prefer so-called "connectionist" over "symbolic" computer implementations or vice versa.[2] Our time is better spent developing and testing algorithms than arguing about these points.

[2] Comprehensibility of the knowledge base, which favors symbolic representations, is a separate issue.

5 Knowledge Refinement

As research continues on the problem of using ML for knowledge acquisition, we will develop more guided approaches than the weak search methods. One step that has already been taken in this direction is that of automatically refining incorrect or partial domain knowledge [4, 11, 12, 15, 16, 20, 21, 22, 26, 30, 31, 32, 33, 34, 47, 50]. Even if we do not have a fully satisfactory set of rules for solving a problem, our learning algorithms can still benefit from the incomplete knowledge we do have. Knowledge refinement systems such as those cited are often able to use partial knowledge to produce better solutions to real-world problems than was previously possible with weak methods alone.

Knowledge refinement systems, like other learning systems, can be symbolic or connectionist. A symbolic approach typically starts with a set of imperfect rules from a human expert and iteratively modifies it in order to improve its correctness or coverage, e.g. by adding and deleting terms. EITHER [32, 33] and NEITHER [4] are systems which refine propositional Horn clause rule sets in such a manner, and FORTE [26] extends the technique to function-free Horn clause representations of logic programs.

The KBANN family of algorithms represents a connectionist knowledge refinement approach. It translates a set of propositional rules [50] or a description of a finite state automaton [20] into the nodes and weights of an ANN. The network, and therefore its embodied knowledge, is then refined by standard ANN backpropagation training [41]. One can then either use the modified network as it stands or apply methods to extract symbolic rules from it [10, 13, 14, 42, 48, 49].

Knowledge refinement systems can take advantage of partial knowledge and correct and embellish it automatically through ML techniques. Their use will greatly reduce the effort needed to create knowledge bases for intelligent systems.

6 Are Rules Sufficient?

There is a possibility that some problems simply cannot be solved by symbolic rules. Perhaps the reason human cognitive processes are so hard to pin down is that they operate in a fundamentally distributed and unrule-like way. Chaos theory tells us the only accurate model of the weather is the weather itself. The idea that the world is its own best model has sometimes been used to argue against knowledge representation in any form [2, 3, 5, 6]. Perhaps the only way to model human cognition is through a device that is similar in structure and complexity to the human brain [27]. Penrose [36] suspects that the physics of brain operation makes some of our thought processes (especially the feeling of awareness) nonalgorithmic, questioning the "strong AI" position that all our thinking is merely the enacting of some algorithm. If this is true, we may have no hope of modeling these aspects at all, either by symbolism or connectionism, using current computer architectures.

I would like to challenge the extremity of these positions. Though it is true that we cannot precisely model the weather at a micro scale, this does not mean there is no high-level structure amenable to abstraction. A meteorologist does not need to predict the temperature of every cubic centimeter of air to tell us it will drop when a cold front moves in. This is a simple symbolic rule with real predictive power.

In chaotic domains, any model at all—symbolic or not—must approach the complexity of the system itself in order to achieve arbitrary accuracy, but this misses the point of having a model in the first place. One needs only a very small set of rules to do better than chance in predicting the next day's weather. One of the simplest and most accurate systems for one-day weather forecasting consists of a single rule: "Tomorrow's weather will be the same as today's." Simplification through models allows us to find order and understanding where there would otherwise be none.

In this vein, symbolic rules may be used to model the processes of cognition, even though the brain's implementation is a distributed one. Much of our thinking can be described symbolically. We communicate with one another with symbols, and we store knowledge in external libraries and other media in the same way. There is thus plenty of reason to expect rule-driven symbol manipulation à la the classic *Physical Symbol System Hypothesis* [29] to be a reasonable model for many aspects of human intelligence. Just as we need not reproduce every detail of bird anatomy to make an airplane that flies, we need not reproduce every cell and connection of the brain to make a machine that thinks. I believe symbolic rules are sufficient to capture most aspects of human intelligence at the everyday level of granularity most useful to us, even though at a micro level they will operate differently than the human brain.

7 Are Emotions Necessary for Learning?

Whether we need to include emotions in our learning systems may seem like a strange question, but with a moment's thought we realize that much of human learning is motivated by emotions. Our engineer of Section 4 spoke of emotions as if they were totally irrelevant to machine intelligence. However, the same cannot be said of human intelligence. Children must receive love and nurture to survive and thrive. Emotional involvement is a powerful motivator in their development and success and continues to be throughout adulthood. In a classic essay, Hadamard [18] investigates the role of human emotion in fostering creative discovery and invention. If we hope to build truly intelligent machines, might we not also need to build in such a motivating drive? Even if it is not completely necessary for artificial systems, can we afford to ignore this complex and powerful urge to learn?

In *The Society of Mind* [24], Minsky casts emotions as fundamental to the success of our intelligence. They spur our creativity while preventing us from obsessively fixating on a single idea or purpose. Without them, we would become robotic drones and accomplish little. Emotions are important checks and balances in the complex system of mind. However, Minsky does not attribute any special status to emotions. He views them simply as tools that interacting mental agents use to accomplish their goals. For example, he describes *Anger* as a tool agent *Work* can exploit to prevent agent *Sleep* from gaining control of the mind. No mysterious qualities need be assigned to *Anger* to explain it. It is simply one of many competing mechanisms which help get things done in the mind.

Newell [28], on the other hand, defines intelligence without reference to emotions. For Newell, intelligence depends only on how well a system uses the knowledge it has. Perfect use constitutes perfect intelligence, while a system that ignores its knowledge has no intelligence.

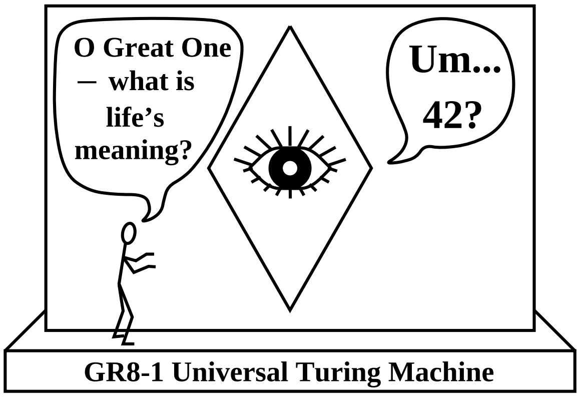
I find Newell's definition flawed specifically where emotions and learning are concerned. A system with emotions may have a curiosity that leads it to formulate and test theories about the world. It does not know whether these theories are true, nor does it know it may benefit from testing them, so any exploration and learning arising from this curiosity do not count in Newell's definition of intelligence. An otherwise identical system that lacks the motivation of curiosity, and so learns nothing, is considered equally intelligent. Nonetheless, empirical investigation often leads to new knowledge that can improve life for the system. Would we not credit a curious, exploring, experimenting system that continually expands its own knowledge base, capabilities, and efficiency (and happiness, perhaps?) with more intelligence than a mentally sedentary one that mechanically applies the same old knowledge to just get by? I hope we would!

Does this imply that emotions are *necessary* for learning? Not at all. While emotions play a key role in motivating human learning, they are certainly not the only possible incentives for learning in general. One may sharpen a skill simply by repeating a task many times, whether one intends to become better at it or not. One may make a great discovery purely by accident. Furthermore, computers are not humans, and they can be motivated in other ways. In a computer system, learning may simply be something the machine is required to do by its program. Both emotions and learning should be important components of any definition of intelligence, but emotions are not prerequisite for learning to occur.

8 Building Superintelligences

Most of the time the ultimate goal of AI is stated as building an artificial intelligence of human capabilities, as suggested by the famous *Turing test* [51]. As long as we are being ambitious, however, why not aim for intelligences that are even greater? Why stop with a machine Albert Einstein if we can hope for even more? Even though this is far beyond our present capabilities, it should still be a subject to think about (Figure 3).

Figure 3:


O Great One — what is life's meaning?
Um... 42?
GR8-1 Universal Turing Machine

Assuming we had already reached the goal of creating machines as smart as individual humans, what would be our next step toward the higher goal of superintelligences? One avenue to explore is that of societies of intelligent agents. We could seek emergent superintelligence from the interactions of "regular" intelligences in much the same way Minsky seeks emergent intelligence from the interactions of unintelligent agents in *The Society of Mind* [24]. This may be a useful insight, but we must examine it more closely to reap its potential benefits. To wit, if our Einstein unit (person or machine) has an IQ of 300, do three average people (100 IQ each) equal one Einstein? I doubt it. They are probably more like 0.4 Einsteins. One might therefore argue that we just need ten or so average people to boost up to one Einstein. I don't buy this either—there is surely a law of diminishing returns operating such that each successive person adds progressively less to the Einstein index, even if only due to communications problems.

Does a colony of thousands of micro-Einsteinian ants ever approach an Einstein of intelligence? Probably not, but ants may be a bad example—their interagent communication and knowledge storage capabilities are surely quite limited. Perhaps the only real problem with applying the extended society of mind metaphor to humans is that humans are too loosely coupled (i.e. our communication bandwidth is too low). We can store as much knowledge as we want using external media. The problem is only in how quickly we can process and apply it.

I postulate that sophisticated symbolic communication among intelligent agents is sufficient to achieve emergent superintelligence. The main reason we do not see obvious mega-Einsteinian strides in the intelligence of cooperating groups of people is slow communication.

If low bandwidth is the only substantive obstacle in the path of emergent superintelligence in human societies, we should be able to see evidence of mega-Einsteinian accomplishments if we observe societies for a long enough period of time. And lo—this is exactly what we *do* see in the rise of technological societies! The knowledge and achievements of these systems is vastly greater than anything a single human could ever accomplish, no matter how smart. So without even realizing it, we already have hard evidence of the success of this approach to building superintelligences! If we could implement in a machine (or machines) a large number of intelligent agents communicating and interacting mentally at high speeds, we might get somewhere in our fantasy project of producing a time-localized, mega-Einsteinian reasoner.

Since humans will probably not be networking their minds together telepathically any time soon, our best hope for a high-speed, superintelligent reasoning system is to build an artificial one. Interacting conglomerations of intelligent agents present a realistic paradigm for achieving this. In the mean time, we should reexamine the idea of human-level intelligence emerging from collections of interacting unintelligent agents. I believe this is the most likely route to our first truly intelligent machine.

9 Conclusion

We have explored some important current issues of knowledge and learning for the creation of artificial intelligence, raising many questions and, hopefully, a few answers in the process. If my presentation has also raised a few eyebrows, so much the better. I believe that knowledge and learning are both essential to the enormous task of implementing intelligent artificial systems, and research on these fronts is steadily progressing. At the same time, as we toil through the technical details of basic research, we should not lose the ability to dream of greater things for tomorrow. It is these dreams that will make intelligent machines a reality.

10 Acknowledgments

Like most human intellectual achievements, this paper is the product of many brains. I would like to thank those who provoked my thoughts on these subjects through symbolic communication, either spoken or written, especially Larry Travis, who inspired the character of the Philosopher in Section 4; Derek Zahn, who exposed me to different points of view; Marvin Minsky, whose ideas [24] helped mine to germinate; three anonymous reviewers, whose comments and suggestions greatly improved the paper; and, of course, Douglas Adams, whose work [1] elicited Figure 3.

References

[1] Douglas Adams. *The Hitchhiker's Guide to the Galaxy.* Harmony Books, New York, NY, 1980.

[2] P.E. Agre and D. Chapman. Pengi: An implementation of a theory of activity. In *Proc. 6th Nat'l Conf. on Artificial Intelligence*, pages 268–272, Seattle, WA, 1987.

[3] P.E. Agre and D. Chapman. What are plans for? *Robotics and Autonomous Systems*, 6:17–34, 1990.

[4] P.T. Baffes and R.J. Mooney. Symbolic revision of theories with M-of-N rules. In *Proc. 13th Int'l Joint Conf. on Artificial Intelligence*, pages 1135–1140, Chambéry, Savoie, France, 1993. Morgan Kaufmann.

[5] R.A. Brooks. Elephants don't play chess. *Robotics and Autonomous Systems*, 6:3–15, 1990.

[6] R.A. Brooks. Intelligence without representation. *Artificial Intelligence*, 47:139–159, 1991.

[7] M.C. Burl, U.M. Fayyad, P. Perona, P. Smyth, and M.P. Burl. Automating the hunt for volcanoes on Venus. In *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition: Proc.*, Seattle, WA, 1994. IEEE Computer Society Press.

[8] K.J. Cherkauer and J.W. Shavlik. Protein structure prediction: Selecting salient features from large candidate pools. In *Proc. 1st Int'l Conf. on Intelligent Systems for Molecular Biology*, pages 74–82, Bethesda, MD, 1993. AAAI Press.

[9] K.J. Cherkauer and J.W. Shavlik. Selecting salient features for machine learning from large candidate pools through parallel decision-tree construction. In H. Kitano and J.A. Hendler, editors, *Massively Parallel Artificial Intelligence*, pages 102–136. AAAI Press/MIT Press, Menlo Park, CA/Cambridge, MA, 1994.

[10] M.W. Craven and J.W. Shavlik. Using sampling and queries to extract rules from trained neural networks. In *Machine Learning: Proc. 11th Int'l Conf.*, pages 37–45, New Brunswick, NJ, 1994. Morgan Kaufmann.

[11] S.K. Donoho and L.A. Rendell. Rerepresenting and restructuring domain theories: A constructive induction approach. *Journal of Artificial Intelligence Research*, 2:411–446, 1995.

[12] L.M. Fu. Integration of neural heuristics into knowledge-based inference. *Connection Science*, 1(3):325–340, 1989.

[13] L.M. Fu. Rule learning by searching on adapted nets. In *Proc. 9th Nat'l Conf. on Artificial Intelligence*, pages 590–595, Anaheim, CA, 1991. AAAI Press.

[14] S.I. Gallant. *Neural Network Learning and Expert Systems.* MIT Press, Cambridge, MA, 1993.

[15] A. Ginsberg. *Automatic Refinement of Expert System Knowledge Bases.* Pitman, 1988.

[16] A. Ginsberg. Theory reduction, theory revision, and retranslation. In *Proc. 8th Nat'l Conf. on Artificial Intelligence*, pages 777–782, Boston, MA, 1990. AAAI Press/MIT Press.

[17] R.V. Guha and D.B. Lenat. Cyc: a midterm report. *AI Magazine*, 11(3):32–59, 1990.

[18] J. Hadamard. *An Essay on the Psychology of Invention in the Mathetmatical Field.* Princeton University Press, Princeton, NJ, 1945.

[19] D. Lenat, M. Prakash, and M. Shepherd. CYC: using common sense knowledge to overcome brittleness and knowledge acquisition bottlenecks. *AI Magazine*, 6(4):65–85, 1986.

[20] R. Maclin and J.W. Shavlik. Refining domain theories expressed as finite-state automata. In *Machine Learning: Proc. 8th Int'l Wkshp.*, pages 524–528, Evanston, IL, 1991. Morgan Kaufmann.

[21] R. Maclin and J.W. Shavlik. Creating advice-taking reinforcement learners. *Machine Learning*, 22(1–3), 1996.

[22] J.J. Mahoney and R.J. Mooney. Combining connectionist and symbolic learning to refine certainty-factor rule-bases. *Connection Science*, 5(3–4):339–364, 1993.

[23] R.S. Michalski. Understanding the nature of learning: Issues and research directions. In R.S. Michalski, J.G. Carbonell, and T.M. Mitchell, editors, *Machine Learning: An Artificial Intelligence Approach, Volume II*, pages 3–25. Morgan Kaufmann, San Mateo, CA, 1986.

[24] M. Minsky. *The Society of Mind.* Simon and Schuster, New York, NY, 1986.

[25] T.M. Mitchell and S.B. Thrun. Explanation-based neural network learning for robot control. In *Advances in Neural Information Processing Systems*, volume 5, Denver, CO, 1993. Morgan Kaufmann.

[26] R.J. Mooney and B.L. Richards. Automated debugging of logic programs via theory revision. In *Proc. Second Intl. Wkshp. on Inductive Logic Programming*, Tokyo, Japan, 1992.

[27] J. von Neumann. *The Computer and the Brain.* Yale University Press, New Haven, CT, 1958.

[28] A. Newell. *Unified Theories of Cognition.* The William James Lectures, 1987. Harvard University Press, Cambridge, MA, 1990.

[29] A. Newell and H.A. Simon. Computer science as empirical inquiry: Symbols and search. *Communications of the ACM*, 19(3), 1976.

[30] D.W. Opitz and J.W. Shavlik. Heuristically expanding knowledge-based neural networks. In *Proc. 13th Int'l Joint Conf. on Artificial Intelligence*, volume 2, pages 1360–1365, Chambéry, Savoie, France, 1993. Morgan Kaufmann.

[31] D.W. Opitz and J.W. Shavlik. Using genetic search to refine knowledge-based neural networks. In *Machine Learning: Proc. 11th Int'l Conf.*, pages 208–216, New Brunswick, NJ, 1994. Morgan Kaufmann.

[32] D. Ourston and R.J. Mooney. Changing the rules: A comprehensive approach to theory refinement. In *Proc. 8th Nat'l Conf. on Artificial Intelligence*, pages 815–820, Boston, MA, 1990. AAAI Press/MIT Press.

[33] D. Ourston and R.J. Mooney. Theory refinement combining analytical and empirical methods. *Artificial Intelligence*, 66(2):273–309, 1994.

[34] M. Pazzani and D. Kibler. The utility of knowledge in inductive learning. *Machine Learning*, 9(1):57–94, 1992.

[35] E.P.D. Pednault. Some experiments in applying inductive inference principles to surface reconstruction. In *Proc. 11th Int'l Joint Conf. on Artificial Intelligence*, pages 1603–1609, Detroit, MI, 1989. Morgan Kaufmann.

[36] R. Penrose. On the physics and mathematics of thought. In R. Herken, editor, *The Universal Turing Machine: A Half-Century Survey*, pages 491–522. Oxford University Press, Oxford, England, 1988.

[37] D.A. Pomerleau. Efficient training of artificial neural networks for autonomous navigation. *Neural Computation*, 3:88–97, 1991.

[38] J.R. Quinlan. Learning logical definitions from relations. *Machine Learning*, 5:239–266, 1990.

[39] J.R. Quinlan and R.M. Cameron-Jones. Oversearching and layered search in empirical learning. In *Proc. 14th Int'l Joint Conf.*

on Artificial Intelligence, volume 2, pages 1019–1024, Montréal, Québec, Canada, 1995. Morgan Kaufmann.

[40] B. Rost and C. Sander. Prediction of protein secondary structure at better than 70% accuracy. *Journal of Molecular Biology*, 232:584–599, 1993.

[41] D.E. Rumelhart, G.E. Hinton, and R. Williams. Learning internal representations by error propagation. In D.E. Rumelhart and J.L. McClelland, editors, *Parallel Distributed Processing*, volume 1, pages 318–363. MIT Press, Cambridge, MA, 1986.

[42] K. Saito and R. Nakano. Medical diagnostic expert system based on PDP model. In *Proc. IEEE Int'l Conf. on Neural Networks*, pages 255–262, San Diego, CA, 1988. IEEE Press.

[43] H. Schnelle. Turing naturalized: Von neumann's unfinished project. In R. Herken, editor, *The Universal Turing Machine: A Half-Century Survey*, pages 539–559. Oxford University Press, Oxford, England, 1988.

[44] T.J. Sejnowski and C.R. Rosenberg. Parallel networks that learn to prounounce English text. *Complex Systems*, 1:145–168, 1987.

[45] D.B. Skalak and E.L. Rissland. Inductive learning in a mixed paradigm setting. In *Proc. 8th Nat'l Conf. on Artificial Intelligence*, Boston, MA, 1990. AAAI Press/MIT Press.

[46] W.N. Street, O.L. Mangasarian, and W.H. Wolberg. An inductive learning approach to prognostic prediction. In *Machine Learning: Proc. 12th Int'l Conf.*, pages 522–530, Tahoe City, CA, 1995. Morgan Kaufmann.

[47] K. Thompson, P. Langley, and W. Iba. Using background knowledge in concept formation. In *Machine Learning: Proc. 8th Int'l Wkshp.*, pages 554–558, Evanston, IL, 1991. Morgan Kaufmann.

[48] S.B. Thrun. Extracting provably correct rules from artificial neural networks. Technical Report IAI-TR-93-5, Institut für Informatik III, Universität Bonn, 1993.

[49] G.G. Towell and J.W. Shavlik. Extracting refined rules from knowledge-based neural networks. *Machine Learning*, 13(1):71–101, 1993.

[50] G.G. Towell and J.W. Shavlik. Knowledge-based artificial neural networks. *Artificial Intelligence*, 70(1,2):119–165, 1994.

[51] A.M. Turing. Computing machinery and intelligence. *Mind*, 59:433–460, 1950.