

# Future of AI Technology

Marvin L. Minsky

(Toshiba Review, Vol.47, No.7, July 1992.)

People often complain that AI is not developing as well as expected. They say, "Progress was quick in the early years of AI, but now it is not growing so fast." I find this funny, because people have been saying the same thing as long as I can remember. In fact we are still rapidly developing new useful systems for recognizing patterns and for supervising processes. Furthermore, modern hardware is so fast and reliable that we can employ almost any programs we can create. Good new systems appear every year, for different "expert" applications.

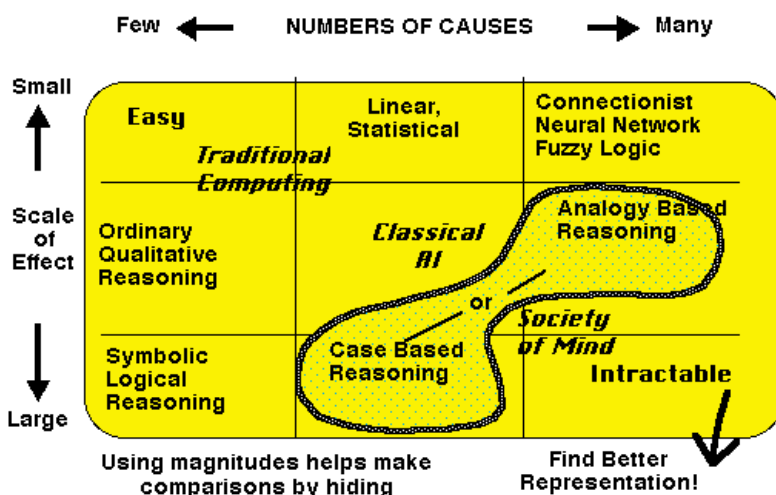
However, progress has been slow in other areas, for example, in the field of understanding natural language. This is because our computers have no access to the meanings of most ordinary words and phrases. To see the problem, consider a word like "string" or "rope." No computer today has any way to understand what those things mean. For example, you can pull something with a string, but you cannot push anything. You can tie a package with string, or fly a kite, but you cannot eat a string or make it into a balloon. In a few minutes, any young child could tell you a hundred ways to use a string -- or not to use a string -- but no computer knows any of this. The same is true for ten thousand other common words. Every expert must know such things.

This is why our "expert systems" cannot communicate. We have programs to play chess, and programs to recognize characters. But no chess program can read text, and no OCR system can play chess. It is almost impossible today to get any two different programs to cooperate. I think that this is because we have not yet developed any systems that can use "common sense." This involves several different kinds of problems. Once we solve them, we will all benefit from the great AI systems of the future.

The first problem of common sense is how to build up that "commonsense knowledge-base." It will have to contain information about strings, roads, tools, energy, books, houses, clothing -- all the knowledge that most children know. This will be hard because much of that knowledge is so "obvious" that people don't need to have words for them. Also, our future commonsense systems will need to understand the "functions" or "uses" of all those things because, to solve real problems, our machine must know which tools or ideas are useful for each kind of purpose. This also means that those knowledge-machines will need to know a lot about human psychology, to understand what kinds of solutions are acceptable. Some research is already under way in this area. One such project is at CM in the United States, and there are other attempts in Japan. Generally, there is so little research in this area that the best applications of AI are still waiting.

Another problem is that you cannot put knowledge into a computer until you find a way to "represent" that knowledge. Technically speaking, for each fragment of knowledge we must first choose some sort of "data-structure" or other way to build the knowledge into memory. It is here that I believe the Computer Science community has used good reasons to make bad decisions! Everywhere I go I find people arguing about which representation to use. One person says, "It is best to use Logic." The next person says, "No, logic is too inflexible. Use Neural Networks." The third person says, "No, Neural Nets are even less flexible, because you have to reduce everything to mere numbers. Instead, you should use Semantic Networks. Then, the different kinds of things can be linked by concepts instead of mere numbers!" But then the first person might complain, "No, Semantic Nets are too arbitrary and undefined. If you use Formal Logic, that will remove those ambiguities."

What is the answer? My opinion is that we can make versatile AI machines only by using several different kinds of representations in the same system! This is because no single method works well for all problems; each is good for certain tasks but not for others. Also different kinds of problems need different kinds of reasoning. For example, much of the reasoning used in computer programming can be logic-based. However, most real-world problems need methods that are better at matching patterns and constructing analogies, making decisions based on previous experience with examples, or using types of explanations that have worked well on similar problems in the past. How can we encourage people to make systems that use multiple methods for representing and reasoning? First we'll have to change some present-day ideas. For example, many students like to ask, "Is it better to represent knowledge with Neural Nets, Logical Deduction, Semantic Networks, Frames, Scripts, Rule-Based Systems or Natural Language?" My teaching method is to try to get them to ask a different kind of question. "First decide what kinds of reasoning might be best for each different kind of problem -- and then find out which combination of representations might work well in each case." A trick that might help them to start doing this is to begin by asking, for each problem, "How many different factors are involved, and how much influence does each factor have?" This leads to a sort of "theory-matrix."



When there only a few causes, each with small effects, the problem will be trivial.

When there are many causes, each with a small effect, then statistical methods and neural networks may work well.

But when we have only a few large-effect causes, that might be a good domain for logical and symbolic problems.

Then, between those extremes, we may be able to reason by using semantic network. Or we might be able to recognize the important patterns by using techniques like Toshiba's multiple similarity methods, or other ways to discover which combinations of features are the most important ones, when there are too many possibilities to explore them all.

When there too many causes with large effects, problems tend to become intractable. Yet even when that happens, we may be able to find useful solutions by using learning and analogy.

In the field of theoretical physics, the researchers have a well- founded hope to discover a "theory of everything" that has some simple and elegant form -- like one of those "unified" theories. However, AI faces an "everything" that includes much more than the physicists face. Because of this, we cannot expect always to simplify and unify our knowledge. Instead, we shall have to learn how to combine and organize our ever-growing theories -- and we ought to undertake that adventure with a positive view. Instead of envying the physicists, we could recognize that ours is a harder and, ultimately, an even deeper and more interesting problem. Our job is to find out how to make "unified theories" about how to combine non- unified theories.

(Revised 95/07/10)

---

Last modified: Wed Jan 02 1997