

The Age of Intelligent Machines: Thoughts About Artificial Intelligence

by [Marvin Minsky](#)

One of the visionaries in the field of AI shares his thoughts on AI, from the beginning of the last decade. From Ray Kurzweil's revolutionary book *The Age of Intelligent Machines*, published in 1990.

What is [Intelligence](#)?

What is [intelligence](#), anyway? It is only a word that people use to name those unknown processes with which our brains solve problems we call hard. But whenever you learn a skill yourself, you're less impressed or mystified when other people do the same. This is why the meaning of "[intelligence](#)" seems so elusive: it describes not some definite thing but only the momentary horizon of our ignorance about how minds might work. It is hard for scientists who try to understand [intelligence](#) to explain precisely what they do, since our working definitions change from year to year. But it is not at all unusual for [science](#) to aim at moving targets. [Biology](#) explores the moving frontier of what we understand of what happens inside our bodies. Only a few decades ago the ability of organisms to reproduce seemed to be a deep and complex mystery. Yet as soon as they understood the [elements](#) of how our gene strings replicate themselves, biologists wondered why it took so long to think of such a simple thing. In the same way each era of [psychology](#) explores what we don't then know about processes in our brains.

Then, can we someday build intelligent [machines](#)? I take the answer to be yes in principle, because our brains themselves are [machines](#). To be sure, we still know very little about how brains actually work. There is no [reason](#) for scientists to be ashamed of this, considering that it was only a century ago that we began to suspect that brains were made of separate [nerve](#) cells that acted somewhat like [computer](#) parts and that it is only half a century since we began developing technical ideas for understanding what such [systems](#) could do. These ideas are still barely adequate for dealing with present-day [serial computers](#), which have only thousands of active [components](#), and are not yet robust enough to deal with [systems](#) like those in the [brain](#), which involve trillions of interconnected parts, all working simultaneously.

Nor do we yet know how to make [machines](#) do many of the things that ordinary people do. Some critics maintain that [machines](#) will never be able to do some of those things, and some skeptics even claim to have proved such things. None of those purported proofs actually hold up to close examination, because we are still in the dark ages of scientific [knowledge](#) about such [matters](#). In any case, we have not the slightest grounds for believing that [human](#) brains are not [machines](#). Because of this, both [psychology](#) and [artificial intelligence](#) have similar goals: both seek to learn how [machines](#) could do many things we can't yet make them do.

Why are so many people annoyed at the [thought](#) that [human](#) brains are nothing more than "mere [machines](#)"? It seems to me that we have a problem with the word "[machine](#)" because we've grown up to believe that [machines](#) can behave only in lifeless, mechanical ways. This view is obsolete, because the ways we use the word "[machine](#)" are out of date. For centuries words like "[machine](#)" and "mechanical" were used for describing relatively simple [devices](#) like pulleys, levers, locomotives, and typewriters. The word "[computer](#)" too inherits from the past that

[sense](#) of pettiness that comes from doing dull arithmetic by many small and boring steps. Because of this, our previous experience can sometimes be a handicap. Our preconceptions of what [machines](#) can do date from what happened when we assembled [systems](#) from only a few hundreds or thousands of parts. And that did not prepare us to think about brainlike assemblies of billions of parts. Although we are already building [machines](#) with many millions of parts, we continue to think as though nothing has changed. We must learn to change how we think about phenomena that work on those larger scales.

What Is [Artificial Intelligence](#)?

Even though we don't yet understand how brains perform many mental skills, we can still work toward making [machines](#) that do the same or similar things. "[Artificial intelligence](#)" is simply the name we give to that [research](#). But as I already pointed out, this means that the focus of that [research](#) will keep changing, since as soon as we think we understand one mystery, we have to move on to the next. In fact, [AI research](#) has made enormous [progress](#) in only a few decades, and because of that rapidity, the field has acquired a somewhat shady reputation! This paradox resulted from the fact that whenever an [AI research](#) project made a useful new discovery, that product usually quickly spun off to form a new scientific or commercial specialty with its own distinctive name. These changes in name led outsiders to ask, Why do we see so little [progress](#) in the central field of [artificial intelligence](#)? Here are a few specialties that originated at least in part from [AI research](#) but later split into separate fields and, in some instances, commercial enterprises: [robotics](#), [pattern recognition](#), [expert systems](#), automatic theorem proving, cognitive [psychology](#), word processing, [machine vision](#), [knowledge engineering](#), symbolic applied [mathematics](#), and [computational linguistics](#).

For example, many [researchers](#) in the 1950s worked toward discovering ways to make [machines](#) recognize various sorts of patterns. As their findings were applied to problems involved with vision, speech, and several other areas, those fields evolved their own more distinct techniques, they organized their own technical societies and journals, and they stopped using the term "[artificial intelligence](#)." Similarly, an early concern of [AI](#) was to develop techniques for enabling [computers](#) to understand [human language](#); this spawned a field called [computational linguistics](#). Again, many ideas from [artificial intelligence](#) had a large influence among psychologists, who applied those ideas to their studies of the mind but used the title "cognitive [psychology](#)."

I can illustrate how [AI](#) projects develop by recounting the [research](#) of James Slagle, who, as a graduate student at MIT in 1960, developed a [program](#) to solve [calculus](#) problems; he named it with the initials of "symbolic automatic integration." Although there were many problems that SAINT couldn't solve, it surpassed the performance of average MIT students. When he first approached this subject, most scientists considered solving those problems to require substantial [intelligence](#). But after Slagle's work we had to ask ourselves instead why students take so long to learn to do such basically straightforward things.

How did SAINT solve those problems? It employed about 100 formulas from the domains of [algebra](#) and [calculus](#) and applied to these about a dozen pattern-matching methods for deciding which formula might be most likely to help solve a given problem. Since any particular attempt might fail, the [program](#) had to employ a good deal of trial and error. If one method did not work, the [program](#) automatically went on to try another. Sometimes one of them would work, but frequently a problem was too hard for any single such method to work. The [system](#) was [programmed](#) in that case to proceed on to certain other methods,

methods that attempted to split each hard problem into several simpler ones. In this way, if no particular method worked, SAINT was equipped with a great variety of alternatives.

Now we can make an important point. For years the public has been told, [Computers](#) do only what they're [programmed](#) to do. But now you can see why that's not quite true: We can write [programs](#) that cause the [machine](#) to [search](#) for solutions. Often such [searches](#) produce results that greatly surprise their [programmers](#).

The idea of making [programs search](#) greatly expanded their powers. But it also led to new kinds of problems: [search](#) processes could generate so many possible alternatives that the [programs](#) were in constant danger of getting lost, repeating themselves, or persisting at fruitless attempts that had already consumed large amounts of [time](#). Much [research](#) in the 1960s was focused on finding methods to reduce that sort of fruitless [search](#). Slagle himself experimented with some mathematical theories of how to take into account both how much effort had been spent on each solution attempt and how much apparent [progress](#) had been made. Thus the SAINT [program](#) worked as well as it did, not merely because of its specialized [knowledge](#) about [calculus](#), but also because of other [knowledge](#) about the [search](#) itself. To prevent the [search](#) from simply floundering around, making one random attempt after another, some of the [program's knowledge](#) was applied to recognize conditions in which its other, more specialized [knowledge](#) might be particularly useful.

When SAINT first appeared, it was acclaimed an outstanding example of work in the field of [artificial intelligence](#). Later other workers analyzed more carefully its virtues and deficiencies, and this [research](#) improved our understanding of the [basic nature](#) of those [calculus](#) problems. Eventually ways were found to replace all the trial and error processes in SAINT by methods that worked without any [search](#). The resulting commercial product, a [program](#) called MACSYMA, actually surpassed the abilities of professional mathematicians in this area. But once the subject was so well understood, we ceased to think of it as needing [intelligence](#). This area is now generally seen as belonging no longer to [artificial intelligence](#) but to a separate specialty called symbolic applied [mathematics](#).

[Robotics](#) and [Common Sense](#)

In the 1960s we first began to equip [computers](#) with mechanical hands and television eyes. Our goal was to endow [machines](#) with the sorts of abilities children use when playing with toys and building blocks. We found this much harder to do than expected. Indeed, a scholar of the [history](#) of [artificial intelligence](#) might get a [sense](#) of watching [evolution](#) in reverse. Even in its earliest years we saw [computers](#) playing [chess](#) and doing [calculus](#), but it took another decade for us to begin to learn to make [machines](#) that could begin to act like children playing with building blocks! What makes it easier to design [programs](#) that imitate experts than to make them simulate novices? The amazing answer is, Experts are simpler than novices! To see why it was harder to make [programs](#) play with toys than pass [calculus](#) exams, let's consider what's involved in enabling a [robot](#) to copy simple structures composed of blocks: we had to provide our [robot](#) with hundreds of small [programs](#) organized into a [system](#) that engaged many different domains of [knowledge](#). Here are a few of the sorts of problems this [system](#) had to deal with:

- The relation between the hand and the eye
- Recognizing objects from their visual appearances
- Recognizing objects partially hidden from view

Recognizing relations between different objects
Fitting together three-dimensional shapes
Understanding how objects can support one another to form stable structures
Planning a sequence of [actions](#) to assemble a structure
Moving in [space](#) so as to avoid collisions
Controlling the fingers of a hand for grasping an object

It is very hard for any adult to remember or appreciate how complex are the properties of ordinary physical things. Once when an early version of our block-building [program](#) was asked to find a new place to put a block, it tried to place it on top of itself! The [program](#) could not anticipate how that [action](#) would change the situation. To catalog only enough fragments of [knowledge](#) to enable a [robot](#) to build a simple blocklike house from an unspecified variety of available materials would be an encyclopedic task. College students usually learn [calculus](#) in half a year, but it takes ten times longer for children to master their building toys. We all forget how hard it was to learn such things when we were young.

Expertise and [Common Sense](#)

Many [computer program](#)s already exist that do things most people would regard as requiring [intelligence](#). But none of those [programs](#) can work outside of some very small domain or specialty. We have separate [programs](#) for playing [chess](#), designing transformers, proving [geometry](#) theorems, and diagnosing kidney diseases. But none of those [programs](#) can do any of the things the others do. By itself each lacks the liveliness and versatility that any normal person has. And no one yet knows how to put many such [programs](#) together so that they can usefully communicate with one another. In my book, *The [Society of Mind](#)*, I outline some ideas on how that might be done inside our brains.

Putting together different ideas is just what children learn to do: we usually call this [common sense](#). Few youngsters can design transformers or diagnose renal ailments, but whenever those children speak or play, they combine a thousand different skills. Why is it so much easier for [AI program](#)mers to simulate adult, expert skills than to make [programs](#) perform childlike sorts of commonsense [thought](#)? I suspect that part of the answer lies in the amounts of variety. We can often simulate much of what a specialist does by assembling a collection of special methods, all of which share the same common character. Then so long as we remain within some small and tidy problem world, that specialist's domain of expertise, we need merely apply different combinations of basically similar rules. This high degree of uniformity makes it easy to design a higher-level supervisory [program](#) to decide which method to apply. However, although the "methods" of everyday [thinking](#) may, by themselves, seem simpler than those of experts, our collections of commonsense methods deal with a great many more different types of problems and situations. Consider how many different things each normal child must learn about the simplest-seeming physical objects, such as the peculiarities of blocks that are heavy, big, smooth, dangerous, pretty, delicate, or belong to someone else. Then consider that the child must learn quite different kinds of strategies for handling solids and liquids; strings, tapes, and cloths; jellies and muds as well as things he is told are prohibited, poisonous, or likely to cut or bite.

What are the consequences of the fact that the domain of commonsense [thinking](#) is so immensely varied and disorderly? One problem is simply accumulating so much [knowledge](#). But [AI research](#) also encountered a second, more subtle problem. We had to face the simple fact that in [order](#) for a [machine](#) to behave as though it "knows" anything, there must exist, inside that [machine](#), some sort of

structure to embody or "represent" that [knowledge](#). Now, a specialized, or "expert," [system](#) can usually get by with very few types of what we call [knowledge representations](#). But in [order](#) to span that larger [universe](#) of situations we meet in ordinary [life](#), we appear to need a much larger variety of types of representations. This leads to a second, harder type of problem: [knowledge](#) represented in different ways must be applied in different ways. This imposes on each child obligations of a higher type: they have to learn which types of [knowledge](#) to apply to which kinds of situations and how to apply them. In other words, we have to accumulate not merely [knowledge](#), but also a good deal of [knowledge](#) about [knowledge](#). Now, experts too have to do that, but because commonsense [knowledge](#) is of more varied types, an ordinary person has to learn (albeit quite unconsciously) much more [knowledge](#) about representations of [knowledge](#), that is, which types of representation skills to use for different purposes and how to use them.

If this sounds very complicated, it is because it actually is. Until the last half century we had only simple theories of mind, and these explained only a little of what [animals](#) could do in the impoverished worlds of laboratory experiments. Not until the 1930s did psychologists like [Jean Piaget](#) discover how many aspects of a child's mind develop through complicated processes, sometimes composed of intricate sequences of stagelike periods. We still don't know very much about such [matters](#), except that the mind is much more complex than imagined in older philosophies. In *The [Society of Mind](#)*, I portray it as a sort of tangled-up bureaucracy, composed of many different experts, or as I call them, "agencies," that each develop different ways to represent what they learn. But how can experts using different [languages](#) communicate with one another? The solution proposed in my book is simply that they never come to do it very well! And that explains why [human consciousness](#) seems so mysterious. Each part of the mind receives only hints of what the other parts are about, and no [matter](#) how hard a mind may try, it can never make very much [sense](#) of itself.

[Supercomputers](#) and [Nanotechnology](#)

Many problems we regard as needing cleverness can sometimes be solved by resorting to exhaustive [searches](#), that is, by using massive, raw [computer](#) power. This is what happens in most of those inexpensive pocket [chess computers](#). These little [machines](#) use [programs](#) much like the ones that we developed in the 1960s, using what were then some of the largest [research computers](#) in the world. Those old [programs](#) worked by examining the consequences of tens of thousands of possible moves before choosing one to actually make. But in those days the [programs](#) took so long to make those moves that the concepts they used were discarded as inadequate. Today, however, we can run the same [programs](#) on faster [computers](#) so that they can consider millions of possible moves, and now they play much better [chess](#). However, that shouldn't fool us into [thinking](#) that we now understand the [basic](#) problem any better. There is good [reason](#) to believe that outstanding [human chess](#) players actually examine merely dozens, rather than millions, of possible moves, subjecting each to more [thoughtful](#) analysis.

In any case, as [computers](#) improved in speed and [memory](#) size, quite a few [programming](#) methods became practical, ones that had actually been discarded in the earlier years of [AI research](#). An [Apple](#) desktop [computer](#) (or an Amiga, Atari, [IBM](#), or whatever) can do more than could a typical million-dollar [machine](#) of a decade earlier, yet private citizens can afford to play games with them. In 1960 a million-[bit memory](#) cost a million dollars; today a [memory](#) of the same size (and working a hundred times faster) can be purchased for the price of a good dinner. Some seers predict another hundredfold decrease in size and cost, perhaps in less

than a decade, when we learn how to make each microcircuit ten times smaller in linear size and thus a hundred times smaller in area. What will happen after that? No one knows, but we can be sure of one thing: those two-dimensional chips we use today make very inefficient use of [space](#). Once we start to build three-dimensional microstructures, we might gain another millionfold in density. To be sure, that would involve serious new problems with power, insulation, and heat. For a [futuristic](#) but sensible discussion of such possibilities, I recommend Eric Drexler's [Engines of Creation](#) (Falcon Press, 1986).

Not only have small [components](#) become cheaper; they have also become faster. In 1960 a typical [component](#) required a microsecond to function; today our [circuits](#) operate a thousand times faster. Few optimists, however, predict another thousandfold increase in speed over the next generation. Does this mean that even with decreasing costs we will soon encounter limits on what we can make [computers](#) do? The answer is no, because we are just beginning a new era of parallel [computers](#).

Most [computers](#) today are still serial; that is, they do only one thing at a [time](#). Typically, a [serial computer](#) has millions of [memory elements](#), but only a few of them operate at any moment, while the rest of them wait for their turn: in each cycle of operation, a [serial computer](#) can retrieve and use only one of the items in its [memory](#) banks. Wouldn't it be better to keep more of the [hardware](#) in actual operation? A more active type of [computer architecture](#) was proposed in [Daniel Hillis's Connection Machine](#) (MIT Press, 1986), which describes a way to assemble a large [machine](#) from a large number of very small, [serial computers](#) that operate concurrently and pass messages among themselves. Only a few years after being conceived, [Connection Machines](#) are already commercially available, and they indeed appear to have fulfilled their promise to break through some of the speed limitations of [serial computers](#). In certain respects they are now the fastest [computers](#) in the world.

This is not to say that parallel [computers](#) do not have their own limitations. For, just as one cannot start building a house before the boards and bricks have arrived, you cannot always start work simultaneously on all aspects of solving a problem. It would certainly be nice if we could take any [program](#) for a [serial computer](#), divide it into a million parts, and then get the answer a million times faster by running those parts simultaneously on that many [computers](#) in parallel. But that can't be done, in general, particularly when certain parts of the solution depend upon the solutions to other parts. Nevertheless, this quite often turns out to be feasible in actual practice. And although this is only a guess, I suspect that it will happen surprisingly often for the purposes of [artificial intelligence](#). Why do I think so? Simply because it seems very clear that our brains themselves must work that way.

Consider that [brain](#) cells work at very modest speeds in comparison to the speeds of [computer](#) parts. They work at rates of less than a thousand operations per second, a million times slower than what happens inside a modern [computer circuit chip](#). Could any [computer](#) with such slow parts do all the things that a person can do? The answer must lie in parallel [computation](#): different parts of the [brain](#) must do many more different things at the same [time](#). True, that would take at least a billion [nerve](#) cells working in parallel, but the [brain](#) has many times that number of cells.

[AI](#) and the World of the [Future](#)

Intelligent [machines](#) may be within the technological reach of the next century. Over the next few generations we'll have to face the problems they pose. Unless some unforeseen obstacles appear, our mind-[engine](#)ering skills could grow to the point of enabling us to construct accomplished artificial scientists, artists, composers, and personal companions. Is [AI](#) merely another advance in [technology](#), or is it a turning point in [human evolution](#) that should be a focus of discussion and planning by all mankind? The prospect of intelligent [machines](#) is one that we're ill prepared to think about, because it raises such unusual moral, social, artistic, philosophical, and religious issues. Are we obliged to treat [artificial intelligences](#) as [sentient](#) beings? Should they have rights? And what should we do when there remains no real need for honest work, when artificial workers can do everything from mining, fanning, [medicine](#), and manufacturing all the way to house cleaning? Must our lives then drift into pointless restlessness and all our social schemes disintegrate?

These questions have been discussed most [thought](#)fully in the literary works of such writers as [Isaac Asimov](#), Gregory Benford, [Arthur C. Clarke](#), Frederick Pohl, and Jack Williamson, who all tried to imagine how such presences might change the aspirations of humanity. Some optimistic [futurists](#) maintain that once we've satisfied all our worldly needs, we might then turn to the worlds of the mind. But consider how that enterprise itself would be affected by the presence of those artificial mindlike entities. That same [AI technology](#) would offer ways to modify the [hardware](#) of our brains and thus to endlessly extend the mental worlds we could explore.

You might ask why this essay mixes both [computers](#) and [psychology](#). The [reason](#) is that though we'd like to talk about making intelligent [machines](#), people are the only such [intelligence](#) we can imitate or study now. One trouble, though, is that we still don't know enough about how people work! Does this mean that we can't develop smart [machines](#) before we get some better theories of [psychology](#)? Not necessarily. There certainly could be ways to make very smart [machines](#) based on principles that our brains do not use, as in the case of those very fast, dumb [chess machines](#). But since we're the first very smart [machines](#) to have evolved, we just might represent one of the simplest ways!

But, you might object, there's more to a [human](#) mind than merely intellect. What about [emotion](#), [intuition](#), courage, inspiration, [creativity](#), and so forth. Surely it would be easier simply to understand [intelligence](#) than to try to analyze all those other aspects of our personalities! Not so, I maintain, because traditional distinctions like those between [logic](#) and [intuition](#), between intellect and [emotion](#), unwisely try to separate [knowledge](#) and meaning from purpose and intention. In [The Society of Mind](#), I argue that little can be done without combining [elements](#) of both. Furthermore, when we put them together, it becomes easier, rather than harder, to understand such [matters](#), because, though there are many kinds of questions, the answers to each of them illuminate the rest.

Many people firmly believe that [computers](#), by their [nature](#), lack such admirable [human](#) qualities as imagination, sympathy, and [creativity](#). [Computers](#), so that opinion goes, can be only logical and literal. Because they can't make new ideas, intelligent [machines](#) lie, if at all, in [futures](#) too remote for concern. However, we have to be wary of such words as "[creativity](#)." We may only mislead ourselves when we ask our [machines](#) to do those things that we admire most. No one could deny that our [machines](#), as we know them today, lack many useful qualities that we take for granted in ourselves. But it may be wrong to seek the sources of those qualities in the exceptional performances we see in our cultural heroes. Instead, we

ought to look more carefully at what we ordinary people do: the things we call common sense and scarcely ever consider at all. Experience has shown that science frequently develops most fruitfully once we learn to examine the things that seem the simplest, instead of those that seem the most mysterious.