

40 years of Oregon's
CHANGING LAKES

A reality check on
ARTIFICIAL INTELLIGENCE

A space telescope watches
HOW COMETS DIE

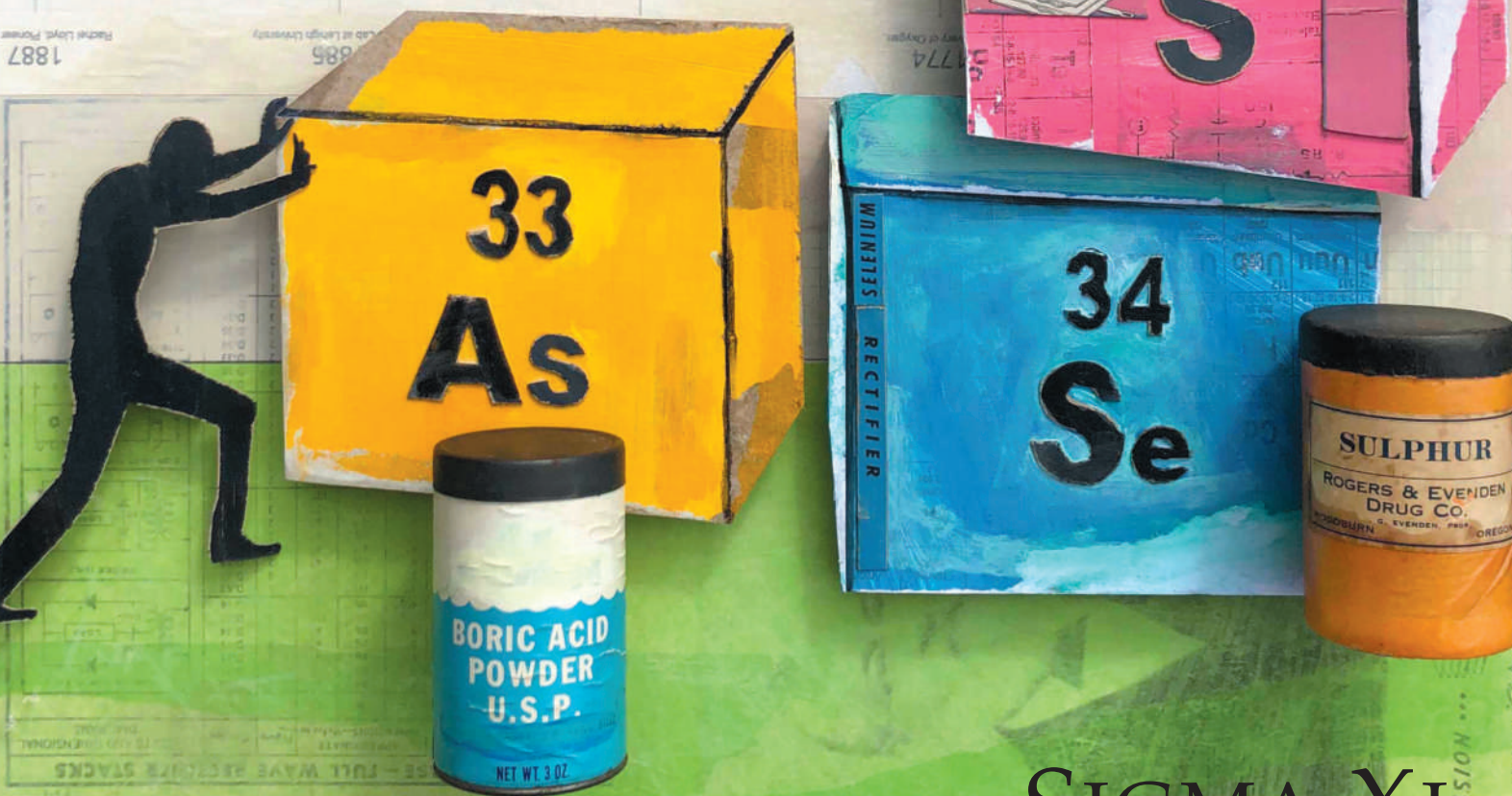
AMERICAN Scientist

November–December 2019

www.americanscientist.org

150 Years of the Periodic Table

Did Sanskrit influence the way
we arrange the elements?



SIGMA XI

Content is copyright protected and provided for personal use only - not for reproduction or retransmission. HONOR SOCIETY
For reprints please contact the Publisher.



Intelligence May Not Be Computable

A hierarchy of artificial intelligence machines ranked by their learning power shows their abilities—and their limits.

Peter J. Denning and Ted G. Lewis

The ultimate goal of the field of artificial intelligence (AI) is to construct machines that are at least as smart as humans at specific tasks. AI has been successful in developing machines that can learn how to recognize speech, find new classes of stars in sky surveys, win grandmaster chess matches, recognize faces, label images, diagnose diseases, hail taxis, drive cars, navigate around obstacles, and much more. Yet none of these machines is the slightest bit intelligent. How can they do intelligent things without being intelligent? Can these machines be trusted when presented with new data they have never seen before? Businesses and governments are using AI in an exploding number of sensitive and critical applications without having a good grasp on when those programs can be trusted.

One way to answer these questions is to classify AI machines according to their relative power, examining what makes machines in each class trustworthy. This way of classifying AI machines gives more insight into the trust question than the more common classifications by the activities and fields in which AI is used.

In its evolution since the 1950s, the AI field has experienced three periods of “boom” punctuated by two periods of “bust” (often called “AI winters”).

Peter J. Denning is Distinguished Professor of Computer Science and Director of the Cebrowski Institute, a hub of information innovation at the Naval Postgraduate School in Monterey, California, and editor of ACM Ubiquity. The author's views expressed here are not necessarily those of his employer or the U.S. Federal Government. Ted G. Lewis is a cofounder of the Center for Homeland Defense and Security at the Naval Postgraduate School. Email for Denning: pjd@nps.edu

The first boom began around the time the field started in 1950. It produced useful prototypes of speech recognizers, language interpreters, game players, math word-problem solvers, and simple robots. But the researchers were not able to deliver on their ambitious claims, and their sponsors pulled back funds in the mid-1970s. Investment returned in the early 1980s, when the Japanese Fifth Generation Project poured large sums into AI research and high-performance logic machines. That boom lasted until the late 1980s, when again the funding agencies were disappointed by lack of progress. The third boom began in the early 1990s with the development of technologies called *machine learning*, which began producing significant, useful, and often surprising results—accompanied by large doses of hype about the future of AI. *Machine learning* refers to programs that develop their function from exposure to many examples, rather than from rules set by programmers. Some AI researchers have placed big bets on this and other methods achieving artificial general intelligence—which may be beyond the reach of machines. If so, a new bust is possible.

An aspect of the hype that has been particularly troubling to us is the claim that all of the advances in computing have come from AI. In fact, computing itself has made steady progress in power and reliability over the past half-century. By 2000, the available computing platforms were sufficiently powerful that they could support AI programs: Modern AI would not exist were it not for the advances in computing, rather than the other way around. Nonetheless, a recent report from the Organization for Economic Cooperation and Development (OECD), a consortium of 34 countries,

defined AI so broadly that any software is a form of AI and that all progress in computing is because of AI. Although that claim is nonsense, it shows the political power that can gather behind hype.

A Learning Hierarchy

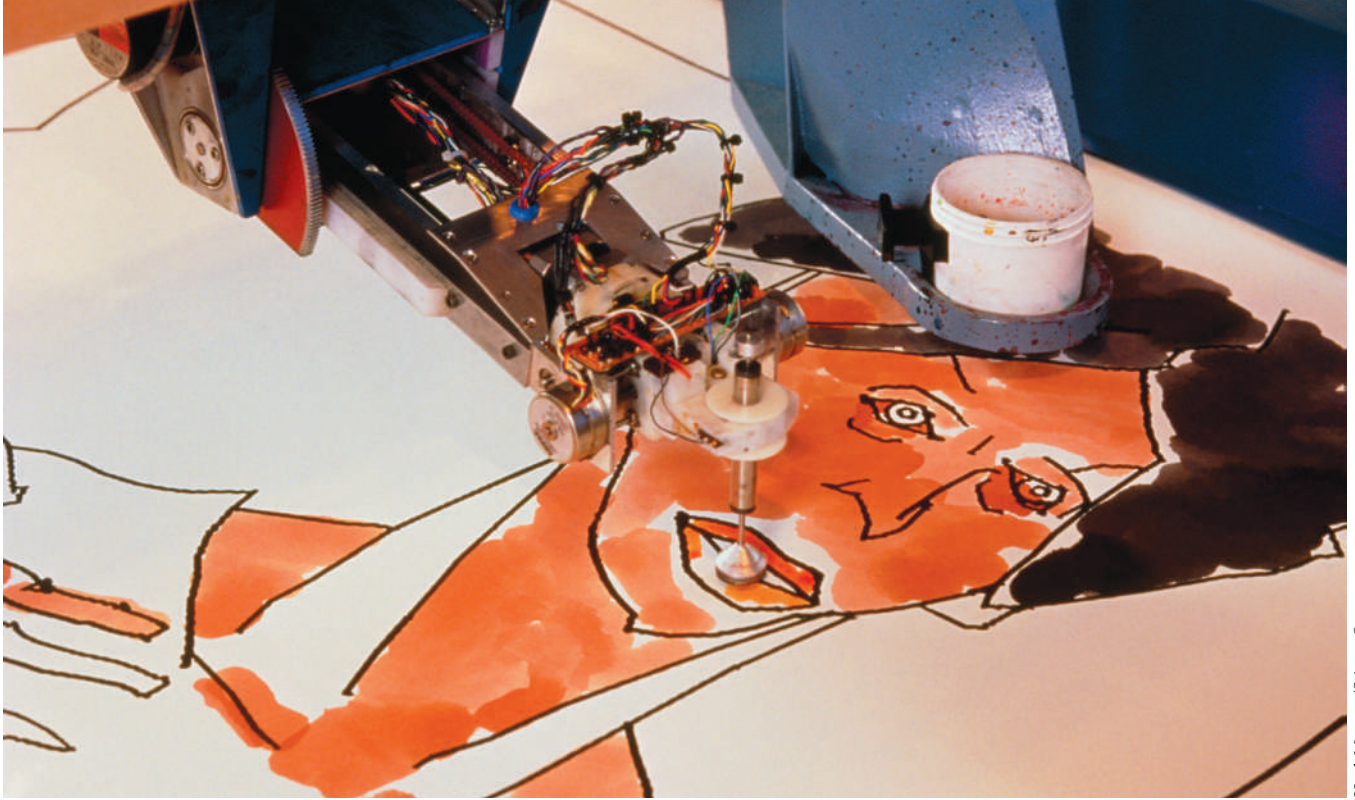
In the task we set for ourselves—classifying these machines and defining their limits—we struggled against two impediments. One is that there is no scientific definition of intelligence. Arthur C. Clarke's admonition—“Any sufficiently advanced technology is indistinguishable from magic”—captures a well-known phenomenon in AI: Once we succeed at building an intelligent machine, we no longer consider it intelligent. As soon as the magic is explained, the allure fades.

The second impediment is our tendency to anthropomorphize—to project our beliefs and hopes about human intelligence onto machines. For example, we believe that intelligent people think fast, yet supercomputers that run a billion times faster than humans are not intelligent.

The hierarchy we will discuss does not rely on any definition of intelligence to classify AI machines. What differentiates the levels of the hierarchy is simply that the machines at a lower level cannot learn functions that higher-level machines can. This grouping is scientifically quantifiable; no anthropomorphizing is needed. This classification does not have to do with computing power. The hierarchy shows that none of the machines so far built have any intelligence at all.

Level 0—Automation

The baseline of the hierarchy is basic automation—designing and implementing automata that carry out or control processes with little or no human



Hank Morgan/Science Source

Artificial intelligence programs (in this case one called AARON) have been used to produce artworks, but whether AI can display creativity when not teamed with a person remains unknown.

intervention. The purpose of automation is to take the human out of the loop by substituting an automaton to do the work. Automation frequently includes simple feedback controls that maintain stable operation by adjusting and adapting to readings from sensors—for example, a computer-controlled thermostat for regulating the temperature of a building, an autopilot, or a factory assembly robot. But the automaton cannot learn any new actions, because its feedback does not change its function to anything beyond what it was built to do, so this kind of automation is not a form of machine intelligence.

Level 1—Rule-Based Systems

Philosophers through the centuries have rated reasoning power as the highest manifestation of human intelligence. AI researchers were attracted to programs capable of imitating the rational reasoning of humans. These were called “rule-based programs” because they made their logical deductions by applying programmed logic rules to their inputs and intermediate results.

Board games were early targets for rule-based programs. Electrical engineer and computing pioneer Arthur Samuel of IBM in 1952 demonstrated a competent checkers program. AI researchers turned their attention to the much harder game of chess, which they thought could be mechanized by

brute-force searching through thousands of future board positions and picking the best moves. That long line of work climaxed in 1997 when an advanced chess program running on an IBM Deep Blue computer beat Garry Kasparov, then the world’s grandmaster at chess. Computer speed was the major reason for this success—the computer can search through billions of moves in the same time a human can search through perhaps hundreds.

Expert systems—programs that solve problems requiring expert-level skill in a domain—were early targets for rule-based programming. Their logical rules are derived from the knowledge of experts. In 1980 John McDermott of Carnegie Mellon University developed

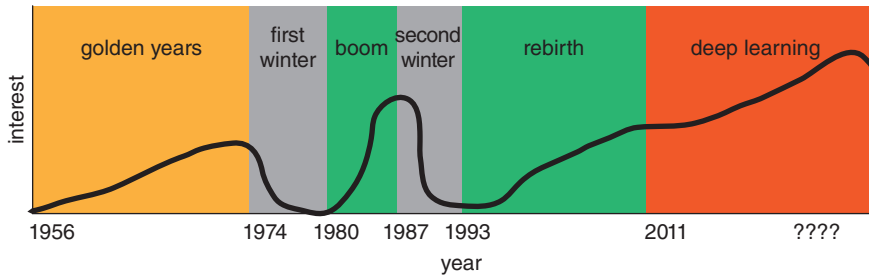
an expert system (called XCON) for the Digital Equipment Corporation. Given customer requirements, XCON recommended configurations of their VAX computer systems, and by 1986 it was reckoned to have saved the company \$25 million annually in labor and facility costs.

But the builders of expert systems soon discovered that getting experts to explain their expertise was often an impossible task: Experts seem to know things that cannot be described as rules. Although a few systems have proved to be competent, no one has built a true expert system.

Level 2—Supervised Learning

At this level, machines compute outputs not by applying logic rules to inputs but by remembering in their structure the proper outputs for each of a set of inputs shown to them by a trainer. The *artificial neural network* (ANN) is a common example, and is so named because its design loosely imitates the structure of a brain, with its many neurons interconnected by axons and dendrites. ANNs consist of many electronic components, called *nodes*, which translate weighted inputs into digital values. The level of interconnectedness, or *weight*, between nodes is determined by the training process. (See figure on page 349.) Natural neural networks from brains were studied by biologists in the late 1800s; artificial neural networks were studied in the 1940s, because some engineers believed that a computer

A Machine-Intelligence Hierarchy	
0	basic automation
1	rule-based systems
2	supervised learning
3	unsupervised learning
4	multiagent interactions
5	creative AI
6	aspirational AI



This timeline shows relative interest in supporting research in AI. The two busts (“AI winters”) resulted from disillusionment in AI progress. The third boom is different because AI is now supported by a large, thriving industry. But another bust is possible if hype isn’t contained.

structured like a brain might be able to perform like a brain.

The trainer of an ANN works with a large set of data consisting of input-output pairs. These pairs are numerous examples of the function the trainer desires the ANN to learn. The outputs are often called *labels*, because a neural network is asked to recognize and label the data presented at the input. For example, the inputs might be bitmaps of photographs of faces, and the outputs would then be the names of the people in the photos. The trainer hopes that the trained ANN will correctly recognize all the faces in the training set. The trainer also hopes that the trained ANN will then correctly recognize faces in photos that were not part of the training set.

The network trainer uses an algorithm called *back propagation* to set the internode connection weights to produce the fewest errors in the output. Even after this process, the errors between actual and desired outputs may not be zero, so there is a chance that the network may produce some incorrect outputs. With their sheer number of nodes and links, ANNs can take days to train, but once trained, they compute their outputs within milliseconds.

Today’s ANNs suffer from two main limitations. One is fragility. When presented with a new (untrained) input, the ANN’s output may deviate significantly from the desired output. Moreover, when a small amount of noise disturbs a valid input, the ANN may label it incorrectly. For example, a road-sign recognizer for a driverless car can be fooled into labeling a stop sign as a speed limit sign simply by placing spots of masking tape at strategic locations on the stop sign. And two ANNs trained from different data samples of the same population may give very different outputs for the same input.

The other ANN limitation is inscrutability. It is very difficult to “explain” how

an ANN reached its conclusion. The only visible result of training an ANN is a gigabyte-sized matrix of connection weights between nodes, so the “explanation” for a problem is diffusely distributed among thousands of weights.

Training ANNs is also expensive, because of the long training process and the expense of getting training sets.

Level 3—Unsupervised Learning

Machines at this level learn to improve their performance by making internal modifications without the assistance of an external training agent. They are attracting more research attention because they can potentially eliminate the large cost of obtaining training sets.

Some AI researchers believe a human-machine team will always outsmart the same machine operating alone.

An early example is a program called AutoClass, built in 1988 by Peter Cheeseman and his colleagues at NASA Ames Research Center. AutoClass computed the most probable classes of 5,425 experimental observations from the NASA infrared telescope; with one exception, it agreed with the classification already determined by astronomers, and the exception was seen as a new discovery.

A recent success is a machine called AlphaGo. Go, an ancient game very popular in Asia, is considered orders of magnitude more difficult than chess. After about six years of development, AlphaGo made a dramatic

debut in 2016, beating Go grandmaster Lee Sedol of South Korea. The AlphaGo machine had been trained by playing against another AlphaGo machine. The two played a massive number of rounds, recording all their moves in each round. When one won a game, it earned a reward, which was propagated back to all the moves that led to the win, thus reinforcing those moves in the next round. At the start, the only prior information given to the machines was the statement of the rules of Go—but no examples of Go games.

AlphaGo was built by the Google subsidiary DeepMind. After their success with Go, they wondered if the AlphaGo platform could be modified to learn chess and another two-player strategy game called Shogi. They renamed their machine AlphaZero to reflect its more general use. Using the same two-machine training method, AlphaZero learned to play grandmaster chess in 9 hours, Shogi in 12 hours, and Go in 13 days. This is a significant breakthrough. AlphaZero’s Go machine accomplished in less than two weeks a feat that no one had done before.

AlphaZero could be used for business games, marketplace games, or war games, with well-defined rule sets describing reward functions, allowable moves, and prohibited moves. But the AlphaZero method may not work with social systems, where the game must be inferred by observing the play.

Level 4—Multiagent Interactions

At this level, machine intelligence emerges from the interactions of thousands or millions of agents, each with a specific function. An agent is an autonomous machine or code segment. The machine-learning capability arises from the collective. This idea was discussed by AI researchers beginning in the 1960s and was the basis of HEARSAY, a speech-recognition system, in the 1970s. It morphed into *blackboard systems* in the 1980s and was epitomized by the late AI pioneer Marvin Minsky in his 1986 book, *The Society of Mind*. A blackboard is a shared knowledge space that agents continually read and update until they converge on a collective solution to a problem.

So far, nothing close to human intelligence has emerged when all of the agents are machines. The story is different when some of the agents are humans. After IBM Deep Blue beat him in 1997, Garry Kasparov invented a

new kind of chess, which he called Advanced Chess, in which a “player” is a team consisting of a human augmented by a computer. It was soon found that the teams of competent players and good chess programs were able to defeat the best machines.

Another example can be found in high school robotics competitions, where teams of human navigators augmented with autonomous function agents are the most frequent winners.

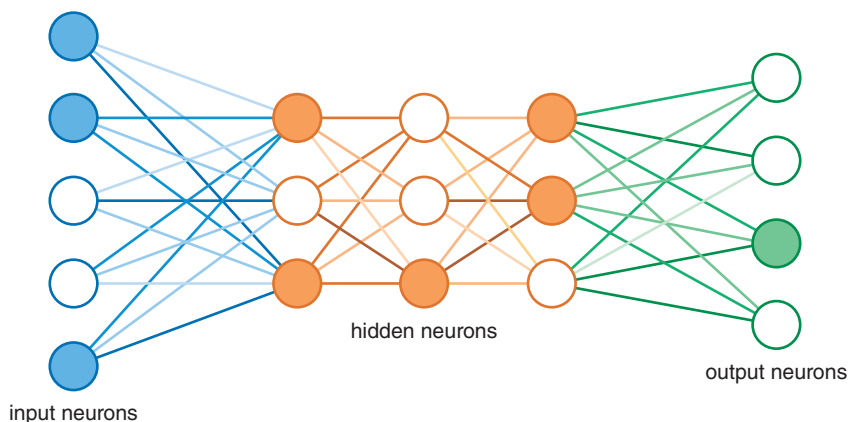
The success of human-machine teams has exposed a rift among AI researchers. Some want machines that are intelligent on their own, with no human assistance. Others believe that a team of a machine and humans can outsmart the same machine operating alone.

Level 5—Creative AI

This level is intermediate between machines that support creative teams and machines that demonstrate general intelligence. The question is: Can there be a machine that is creative on its own without the benefit of a team? At the current stage of the technology, there are no working machines that can perform at either level 5 or 6.

Some AI researchers have speculated that creativity is the recombination of existing ideas, and they have experimented with machines that do that. An example is the genetic algorithm, popularized around 1975 by John H. Holland at the University of Michigan, to find near-optimal solutions to problems by simulating genetic mutation and cross-combination. An early use of this algorithm was in a U.S. Navy robot that could safely find its way through a minefield. Genetic algorithms start out with random strings of instruction, and each is rated with a fitness value based on its demonstrated ability. The programs with the best fitness values are combined to create a new generation of programs that are fitness-rated, and so forth. Through generations of refinement, programs evolve into successes, such as ones that can safely navigate minefields.

Artists and musicians have experimented with AI tools to produce new art forms. The Prisma app, which transforms photographs into art images in the style of famous painters, is an early example. Ahmed Elgammal of Rutgers University and his colleagues have demonstrated works of art generated by a neural network machine called AICAN; his conclusion was that



An artificial neural network (ANN) is a network of electronic nodes arranged in a series of layers, each providing input for the next. The input layer drives one or more hidden layers, the last of which drives an output layer. A node “fires” when the weighted sum of its inputs from the previous layer exceeds a threshold. The weights are parameters that can be adjusted by a training algorithm so that the output for a given input matches the desired output.

although the AI appeared to be artistically creative, it was not as creative as an artist armed with an AI tool (see “AI Is Blurring the Definition of Artist,” Arts Lab, January–February).

Creativity seems to be a deeply social process involving many human assessments about new possibilities and contexts. It may not be possible to build a machine that rises, on its own, to this kind of creativity.

Level 6—Aspirational AI

This level includes a variety of speculative machines that represent the dreams of many AI researchers. The most ambitious dreams feature machines that think, reason, understand, and are self-aware, conscious, self-reflective, compassionate, and sentient. No such machines have ever been built, and no one knows whether they can be built.

Early on, researchers realized that AI machines lacked common sense. Early medical expert systems, for example, were prone to make mistakes no doctor would make. Researchers thought that the solution was to gather a large compendium of common-sense facts and rules in a very large database for use by the expert system. In 1984 Douglas Lenat, CEO of Cyc Corporation, set out to build such a machine, which he called Cyc. His project continues to this day. The machine, which now contains millions of common-sense facts, has never succeeded in helping an expert system behave like a human.

Much of AI research has been predicated on the assumption that the brain is a computer and the mind is its software. Cognitive scientists now believe that the

structure of the brain itself—intricate folds, creases, and cross-connections—gives rise to consciousness as a statistical phenomenon of brain activity. But even further, it appears that much of what we think we know is actually distributed in the social networks of which we are a part; we “recall” it by interacting with others. Chilean biologists Humberto Maturana and Francisco Varela argued that biological structure determines how organisms can interact and that consciousness and thought arise in networks of coordination of actions. A conclusion is that autonomous software and biologically constructed machines will not be sufficient to generate machine intelligence. In ways we still do not understand, our social communities and interactions in language are essential for general intelligence.

Moving the Goals

New applications of AI are announced every day, but AI technology is not yet advancing toward levels 5 and 6. AI is getting better at levels 2 through 4. We need to separate excitement over new or improved applications from true advancement in the power of AI algorithms to solve certain classes of problems.

The hierarchy leads to the tantalizing—though likely unpopular—conclusion that human intelligence is not computable. It may be that the peak of machine AI is to support human-machine teaming. And that is a significant goal in itself.

(References for this article are available online at www.americanscientist.org.)