Algorithms Are Not Enough

Creating General Artificial Intelligence

Herbert L. Roitblat

Algorithms Are Not Enough

Creating General Artificial Intelligence

Herbert L. Roitblat

The MIT Press
Cambridge, Massachusetts
London, England

This book was set in ITC Stone Serif Std and ITC Stone Sans Std by New Best-set Typesetters Ltd.

Library of Congress Cataloging-in-Publication Data

Names: Roitblat, H. L., author.
Title: Algorithms are not enough : creating general artificial intelligence / Herbert L. Roitblat.
Description: Cambridge, Massachusetts : The MIT Press, [2020] | Includes bibliographical references and index.
Identifiers: LCCN 2019046398 | ISBN 9780262044127 (hardback)
Subjects: LCSH: Artificial intelligence.
Classification: LCC Q335 .R65 2020 | DDC 006.3—dc23
LC record available at https://lccn.loc.gov/2019046398

10 9 8 7 6 5 4 3 2 1

Contents

Preface

At least since the 1950s, the idea that it would be possible to soon create a machine that was capable of matching the full scope and level of achievement of human intelligence has been greeted with equal amounts of hype and hysteria. We have now succeeded in creating machines that can solve specific fairly narrow problems with accuracies that meet or exceed those of their human counterparts, but general intelligence continues to elude us. In this book, I want to outline what I think it will take to achieve not just task-specific intelligence, but general intelligence.

Although some people look forward to achieving artificial general intelligence, others fear it, to the point of predicting that a generally intelligent machine will spell the end of human existence. Such a machine would be able to improve itself, their thinking goes, and will quickly pass from equaling human intelligence to far exceeding it. Computers will become so intelligent that humans will be lucky to be kept as pets. At best, the intelligent computers will ignore us; at worst, they will seek to destroy us as pests competing for resources.

Both views are fundamentally untenable. The tools that let us build specialized intelligence are not up to the task of general intelligence. Even if we make new tools that are capable of achieving general intelligence, they will not result in any kind of explosive self-improvement in intelligence. I describe why improvements in machine intelligence will not lead to runaway machine-led revolutions. Improvements in machine intelligence may change the kind of jobs that people do, but they will not spell the end of human existence. There will be no robo-apocalypse.

I have written this book for a nontechnical reader. If I succeeded, you should not have to know much about computers, psychology, or artificial intelligence to read it.

Read this book if you are interested in intelligence, if you want to know more about how to build autonomous machines, or if you are concerned that these machines will someday take over the world in a sudden explosion of technology called "the technological singularity." Hint: they won't.

I hope to convince you that it is possible to create artificial general intelligence, but it is neither so imminent nor so dangerous as some authors would have you believe. It will take a change in perspective, and I have tried to sketch out just what that new perspective is.

This topic is important because hardly a day goes by without a call for some kind of regulation of artificial intelligence, either because it is too stupid (for example, face recognition) or imminently too intelligent to be trusted. Although this is not a book about policy, good policy requires a realistic view of what the actual capabilities of computers are and what they have the potential to become. Conversely, progress in developing artificial general intelligence requires knowledge that we do not have about the nature of intelligence, brains, and the kinds of problems a generally intelligent agent will have to solve.

As Alan Turing said in 1950, "We can only see a short distance ahead, but we can see plenty there that needs to be done."

allows humans to respond to a dynamically changing world without getting lost in thought.

According to John McCarthy's proposal, along with Marvin L. Minsky, Nathaniel Rochester, and Claude Shannon, the goal of the Dartmouth Summer Workshop was to conduct a study toward the creation of a general artificial intelligence that would be able to form abstractions, solve problems, and improve itself. They thought, at the time, that the way to achieve this general intelligence was to describe as precisely as possible the nature of thought and get a machine to simulate it.

According to the participants, the workshop fell short of its lofty goals, but it can still be described as a profound milestone for the field of artificial intelligence. It is also telling that even at this early date, they focused on the kind of tasks that we associate with higher cognitive function. The participants viewed intelligence as rational, deliberate, and goal directed. For example, Allen Newell, John Clifford Shaw, and Herbert Simon were working on a program to prove mathematical theorems. Their Logic Theorist was intended to mimic the problem-solving skills of an adult human being—in this case, an expert mathematician. Their program would eventually prove 38 of the first 52 theorems from chapter 2 of Alfred North Whitehead and Bertrand Russell's book (*Principia Mathematica*). Some of the Logic Theorist proofs were even novel ones.

Herbert Simon is quoted telling a group of graduate students that he and Allen Newell, had over Christmas, "invented a computer program capable of thinking non-numerically, and thereby solved the venerable mind-body problem, explaining how a system composed of matter can have the properties of mind." Their choice of theorem-proving as their demonstration of mind within a computer was fortunate in that the process of theorem proving was already well-defined as a step-by-step process consisting of a small set of actions (for example, symbol substitution) that could be applied to a small set of basic facts or axioms (for example, symbols). The book that they imitated, in fact, was dedicated to proving the basic properties of mathematics, so it largely laid out the axioms and the operations that could be applied to those axioms.

In hindsight, Newell, Shaw, and Simon's work on the Logic Theorist was a small step from the symbolic logic of *Principia*, but at the time, it was a huge leap for computational intelligence. Their approach would have a profound effect on much of the work that came after it for many years. Even

though Whitehead and Russell had laid out the steps for proving their theorems, it is instructive that the Logic Theorist did not always follow their methods. It proved some of the theorems in novel ways. Simon and his colleagues overestimated the importance of that finding, which was also a milestone in the development of computational intelligence, a tendency that is still commonly repeated.

Today we have computer systems that can play games, diagnose disease, and perform other tasks at suprahuman levels. Each breakthrough achievement is heralded as the next step in the evolution of computational intelligence, allegedly bringing systems closer to the goal of general artificial intelligence. If only we had a bit more memory and faster processors, we would at last be able to achieve general intelligence.

Many things have changed over the years since these early developments, but two things have not changed. One is the overreliance on a small set of processes as the necessary and sufficient ones to build a general intelligence. The computers of the 1950s and 1960s were far too slow and too limited to actually produce a full intelligence, so the researchers settled for solving example or "toy" problems. Their mistake lay in thinking that size and speed were the only limits to expanding these systems to fully achieving a humanlike intelligence.

Their other mistake was the belief that the kinds of problems that they were studying were fully representative of the kinds of problems that a general intelligence would have to solve. They focused on toy versions of problems with specifiable steps that are relatively easy to describe and specific solutions that are easy to evaluate. These kinds of problems can be described as "path problems." Solving them requires finding a path through a "space" that consists of all of the "moves" the system could make. Some combination of moves will solve the problem, and the computer's task is to find the specific path through the available moves that does actually solve it. Computational intelligence is the process of finding the set of operations and their order (the path) necessary to solve a problem.

Another way of describing these problems is, in the words of Judea Pearl, as exercises in curve fitting. To paraphrase his view, solving these problems consists of finding a function that maps the available inputs to the desired outputs. It is just a way of formulating statistical predictions. This mapping process can be quite complex, and the number of choices or estimates that go into forming that relationship can be daunting, but that is still the form

taken by all of the current computational intelligence systems out there. But not all problems are like this. Not all problems are path problems.

The progress that has been achieved in computational intelligence, and it has been dramatic, has come from the genius of system designers to formulate systems that are within the capacity of computers to solve. These systems need not, and generally do not, perform the tasks in the same way that people do because computer scientists have figured out how to reduce them to Pearl's kind of estimation task. They may perform specific tasks better than people do, but this is not because they have exceeded human intelligence in that task but because their designers have found other ways to solve those problems that do not require humanlike intelligence. Maytag dishwashers may clean dishes cleaner than I do by hand, but that does not make them any closer to achieving the intelligence of a human restaurant employee.

None of this is to say that machine learning systems that diagnose disease, understand speech, or drive cars are not intelligent, but they are intelligent in a special-purpose way, not in a general way. If we are to get beyond special-purpose intelligence, we will need to solve problems that are not being addressed today. If we want humanlike intelligence, we must figure out a way to construct it from the tools that we have available or we must build new tools. There are some attempts to create general intelligence with current tools, but none of them, so far, has demonstrated any success. Rather, the more promising road is to try to understand and emulate how the only example of general intelligence we have, people, create this intelligence. Ultimately, machine general intelligence may not resemble human general intelligence in its specific methods, but it must resemble it in the range of its capabilities.

The Invention of Human Intelligence

Over thousands of years, we humans have invented ever more complex artificial thinking tools, but natural human intelligence does not seem to have changed much. To the extent that we are more intelligent than our Paleolithic ancestors, it is because we have combined natural intelligence like what they had with artificial intelligence invented over the centuries.

The inventions of language and then eventually writing were probably among the most important tools added to the human intellectual toolbox.

Although some people argue that language is somehow innate, it appears to have emerged somewhere between 100,000 and 50,000 years ago and to have profoundly expanded the capabilities of hominids (Gabora, 2007). Brains with language, as opposed to the same brains without language, have increased capacities to share information, to coordinate activity, and to transfer experience, among others (Clark, 1998). Language, and particularly syntax, was associated with an enormous expansion of the kind of cognitive processes that these early humans could engage.

According to William Calvin, "Words are tools." Calvin goes on to speculate that the prelanguage human may have been capable of words, which could be used in short expressions, but not capable of complex sentences or of talking about the future or the past. These humans may have been capable of some basic kinds of thought, but not capable of structuring those thoughts, and therefore not capable of manipulating images, hypotheses, or possibilities. Since the invention of language, human intellectual capabilities have changed substantially.

Modern humans migrated to Europe about 43,000 years ago. Cave paintings and carved figures from that period (33,000 to 43,000 years ago), along with musical instruments, were found in the Swabian Jura in southern Germany. The Paleolithic cave paintings in Chauvet cave near France's Ardeche River are thought to be 32,000 years old. According to some anthropologists, the structure and detail of these cave paintings imply that the painters enjoyed a relatively sophisticated mental world. The Lascaux paintings in southwestern France are only about 20,000 years old. During this period, humans began to bury their dead, to create clothes, and to develop complex hunting strategies, such as using pit traps to capture prey. In Asia, cave paintings from the Indonesian island Sulawesi are thought to date from about 35,000 years ago. On the Island of Borneo, figurative cave paintings have recently been described that appear to date from about 40,000 years ago. The cave paintings are an indication that the Paleolithic people were capable of symbolic representation of their environments.

Few artifacts of Paleolithic artificial intelligence survive, but among these are structures that appear to be symbolic of their builders' world. These artifacts may have played a role in helping people navigate their world geographically and perhaps spiritually. Some of them, for example, depict constellations that would have been important to navigation. The painters

of cave paintings may have believed that depicting such things as deer and bison would make it easier to hunt those prey.

There is some evidence that Mesolithic (the period starting about 11,000 years ago) people also developed artifacts that are more recognizably computational, such as calendars. Calendars are clearly important to agriculture, but they may also be important to hunter-gatherers—for example, to time the migration of birds and animals or to collect ripe fruits from distant locations that could not be observed directly.

These calendars used notched stones or bones, for example, to notate the passage of astronomical objects, particularly the moon. Larger structures, like Stonehenge in southern England (5,000 years ago), or an even older calendar structure found in Aberdeenshire in Scotland (about 10,000 years ago) were also astronomical calculators. The Aberdeenshire calendar consists of a series of pits dug in the shapes of the moon's phases, arranged in a 164-foot arc. The arc was aligned with a notch in the landscape where the sun would have risen during the winter solstice, allowing the lunar calendar to be corrected each year to match the solar year.

A Neolithic calendar, Newgrange, is in the Boyne Valley, County Meath, of Ireland. Built over 5,000 years ago, it marks the winter solstice using a roof box that allows sunlight to illuminate a buried chamber around the winter solstice.

Humans have gone from painting on cave walls to inventing interplanetary spacecraft because they have, over many generations, developed thinking tools that enable increasingly sophisticated intellectual activity. Among these tools are:

- mathematics (starting about 4,000 years ago)
- logic (about 2,600 years ago)
- algorithms (about 800 years ago)
- digital computers (about 80 years ago)

Each of these inventions enabled many other inventions and discoveries, which further contributed to human intelligence. Without these tools, human thought tends to be incomplete, irrational, and biased. People jump to conclusions based on wishful thinking and incomplete information. Decisions are made on the basis of how easy it is to think of answers rather than on the correctness of those answers.

adopting a more biologically inspired approach to artificial intelligence. Instead of high-level deliberative rules, neural networks employ models that are more like simplified neurons. Instead of operating on symbols, like the words in a language, neural networks use connections among simulated neurons. The widespread use of neural networks, which have now grown into so-called deep learning models, was responsible for a lot of progress in computational intelligence, but it still did not bring us any closer to achieving general intelligence. Neural networks and other forms of machine learning helped to make it more obvious that the practice of AI, as opposed to the aspirations of AI, was complex functions that mapped inputs to outputs. As Hans Moravec and others asserted, it takes a lot more computation to simulate even a simple neural network than to follow a collection of rules, but both of them are still just calculating functions, an opinion shared by Pearl.

The key part of natural intelligence is the apparent ability to construct problem spaces, not just find paths through one that has already been constructed. But natural intelligence also has other properties. Natural intelligence is not concerned with finding the optimal solution to problems. Rather, natural intelligence is willing to jump to conclusions that cannot be "proven" to be correct in any sense of the word.

Rather than being algorithmic as artificial intelligence is, natural intelligence is heuristic. An algorithm is a set of steps that when followed with a particular input will always yield a corresponding output. A heuristic, on the other hand, is more like a rule of thumb. It mostly works, but sometimes it does not. A baby can recognize his or her mother within hours after birth, but a computer learning to identify categories of objects may require several thousand presentations. Take a child to the zoo and buy him cotton candy, and that kid will expect the same treat on all future visits.

In contrast to the intellectual capacities modeled by computational intelligence, many of the basic cognitive functions that I have called natural intelligence are shared by other species. Precocial birds (birds that can feed themselves immediately after hatching), such as chickens and ducks, learn to identify their parents within hours of birth. Scrub jays and other birds can store seeds under rocks and in crevices and recover them even months later after their environment has been covered by snow. As Wolfgang Köhler showed, chimpanzees can solve certain kinds of insight problems. Rather

than learn by trial and error, chimpanzees were observed to put two sticks together or to stack boxes in order to reach food that was otherwise out of their reach.

Many animals, from ants to bears and chimpanzees, have been found to be able to respond to small numerical quantities (typically on the order of one to four or six) when other features have been controlled. Dogs and other animals can learn the names of up to about a thousand objects with some training and can select those objects following verbal commands.

Natural human intelligence or that found in animals can play an important role in that species' cognition. But the full intellectual achievement of humans up to this point has depended on using that native intelligence plus additional thinking tools that have been invented to achieve the current level of intellectual functioning.

Human natural intelligence has mostly been studied in the context of the foibles and failures it produces in educated humans or in the context of psychological development. It has been largely neglected as a source of human achievement, so we know a lot about the biases and limits it imposes on intelligence, but little about the positive contributions it makes. Natural intelligence is extremely likely to play a critical positive role in general human intelligence, and if we can figure it out, likely to play an important role in computational intelligence as well. Humans could not have invented their thinking tools without it and could not function if they were limited to trial-and-error learning as the early psychologists argued, or to the repeated presentation of labeled examples as modern machine learning would suggest.

The General in General Intelligence

Just how general does general intelligence have to be?

Einstein was really successful at theoretical physics. He won the Nobel Prize for his work on the photoelectric effect—which is the basis for how solar cells generate electricity. Arguably, his work on relativity was even more impactful. As smart as he was, though, Einstein was not good at everything. He was not, apparently, a distinguished mathematician, though he used mathematics very effectively. He may have played chess, but he is very unlikely to have been an accomplished go player. I doubt that he would have done well on the television game show *Jeopardy!*.

There are clear differences among people in their ability to learn, understand, create, analyze, interpret, and adapt to their environments. But not all of these abilities are equal. Einstein could play the piano and the violin, but it was doubtful that his skill with these instruments would have compared favorably to that of Itzhak Perlman or Mozart. Yo-Yo Ma is a great cellist, but I don't think that he has any publications in physics journals. Intellectual performance can vary from task to task, from time to time, as well as from person to person. Although there may be correlations among a person's capability on different skills, that is, a person who performs well on some task is likely to perform well on some others (See chapter 2), being brilliant on some tasks does not guarantee that you are brilliant on others.

Intelligence is a complex concept that involves many different kinds of skills. Psychologists have been measuring intelligence for over a century, but they are mainly interested in identifying the differences among people, rather than identifying the mechanisms by which it is produced. The first intelligence tests were designed to detect students who might need special help in school. The goal was to predict the overall aptitude of the person for learning or for other measures of intellectual success. Intelligence tests may include vocabulary assessments, analogies, image manipulation, or reasoning. Each of these has been found to correlate with some measures of success.

Intelligence tests usually include a battery of different subtests, each directed at measuring a specific ability. The idea of general intelligence as a thing comes from the observation that people's performance on these subtests tend to be correlated. If a person does well on a test that requires image rotation, for example, that person is likely to also do well at answering vocabulary questions.

This correlation among subtest performances has been called the "g-factor" for general intelligence. G could indicate the presence of some kind of general intelligence, for example, some people might have more powerful brains than others and so perform well. Alternatively, g may be merely a label for the statistical correlation. Intelligence, in other words, may not actually be all that general; instead it could be that the tests are not that good at isolating specific abilities. Multiple subtests may assess overlapping sets of specialized capabilities.

For example, a test taker who had vision problems might perform poorly over many tests not because that person is dumber than one with better

vision, but because he has trouble reading the questions. People who are anxious might perform poorly on all tests, and those who are calm might perform better on all tests. Test taking may be its own skill. These associated factors may cause correlations without saying anything about general intelligence.

The correlations on the subtests of an intelligence test are not necessarily indicative or performance on real-world activities. Consider the relative skill sets of Albert Einstein and Yo-Yo Ma. Both are brilliant and are successful in their own, nonoverlapping ways. Intellectual superiority in one area does not guarantee superiority in other areas. We will consider the nature of the correlations in the context of intelligence tests in the next chapter. If human intelligence is any kind of example, artificial general intelligence may not, in the end, be quite as general as some people might expect.

Specialized, General, and Superintelligence

Computational intelligence programs so far have mostly involved performance on a single task, such as playing chess, diagnosing brain injuries, answering *Jeopardy!* questions, and the like. Chess playing was once thought to be a prime example of human intellectual capabilities. Chess was thought to be indicative of using strategy, reading the motivations of other people, and engaging in deep analysis of the situation. In this light, solving the problem of playing chess would go a long way toward addressing general intelligence because it would require the solution of so many higher cognitive functions. A chess-playing computer would have to assess its opponent, understand the person's motivations, and analyze the situation.

In fact, in his famous book, *Gödel Escher Bach*, Douglas Hofstadter argued that "there may be programs that beat anyone at chess, but they will not be exclusively chess programs. They will be programs of general intelligence, and they will be just as temperamental as people. "'Do you want to play chess?' No, I'm bored with chess. Let's talk about poetry" (Hofstadter, 1979, 1999, p. 678).

Instead, just the opposite happened. We have computer programs that are able to play chess at a very high levels, but they are incapable of also talking about poetry. The way chess-playing programs have been designed has nothing to do with deep psychological functions or general intelligence.

Rather, these programs depend on a simpler special-purpose method that organizes potential chess moves into a kind of branching tree. Algorithms are available to search among these branches and identify moves that are likely to lead to a successful outcome for the game. Chess developers reduced the problem of choosing chess moves to the simpler problem of selecting from a series of tree branches.

Playing the game go was predicted to be beyond the capacity of computers. Even the kind of approach that was successful for chess would not work for go, because of the huge number of different possible go positions and the number of ways they could be combined make the go tree too complex to evaluate moves in the same way they can be evaluated for chess. However, computer scientists were recently able to build a system that could play go at a world-class level, because they built another special-purpose algorithm.

The knowledge that went into developing programs that play chess or go is valuable for what it tells us about solving other similarly structured problems. Go became possible when the DeepMind team, who developed the program, designed useful heuristics to limit the number of branches that had to be evaluated to choose a move.

Given the reductionist approach to special-purpose computational intelligence, it should not be surprising that computers have not, so far, made much progress in general intelligence. The creation of yet another special-purpose algorithm may be intelligent, but even a collection of every special-purpose algorithm will not get us to a general intelligence.

Computer science has been effective at building hedgehogs, but not yet at building foxes. The ancient Greek poet Archilochus is commonly quoted as saying, "The fox knows many things, but a hedgehog one important thing." Current computational intelligence systems excel at specific tasks, but none of them yet has achieved any level of generality. There is no reason to think that combining special-purpose systems will, even eventually, result in the emergence of a general intelligence. A fox cannot be constructed from a stack of hedgehogs.

General intelligence, even in humans, is an elusive topic. The correlation among subtests could be due to some kind of brain efficiency, but it could also be a purely statistical artifact. If Einstein had a better brain, then maybe he should have been able to do everything better than other people, but his talent was limited. Intelligence in people, as measured by their successes

as a go-playing computer is of no use to playing chess, it is difficult to see how a paper clip–making computer could be of any use to improving the computational intelligence of computers, including itself. They are different problems, and there is no bridging technology available in the current world or in Bostrom's thought experiment that would allow the computer to move from one to the other. It may learn to better navigate the space of paper-clip making, but that space does not include anything about improving computing. There currently is no method that would allow a chess-playing computer to claim boredom with the game and to then direct its efforts to reading poetry. Creating computers with that kind of capability will require approaches that are not being used, or perhaps not even being imagined today.

Superintelligence does not now exist and the current approaches to AI do not provide a path to get to it. Creating a superintelligent AI would require an approach that we have not yet conceived of. That is not to say that it is impossible, but it does say that we are not yet even heading in the right direction to achieve it. New approaches, invented by people, will be needed to achieve that goal.

This book is intended to provide an understanding of what is needed to achieve general intelligence. It is a road map for research, but not yet a report of the outcome of that research.

The current press coverage of artificial intelligence would have you believe that we are on the verge not only of general intelligence but of a runaway superintelligence that will first come for our jobs and then our babies.

Although it is true that computational intelligence is now capable of taking on a large number of tasks that have previously been performed by humans, it is also creating other new jobs that have never been available in the past. It has the potential to disrupt and change many jobs, but it will not destroy the economy in the process, just change it.

The prospects of an exponentially improving superintelligence that will destroy the world, as in Bostrom's paper-clip thought experiment, are zero as well. Machine learning may be speedier on faster processes, but ultimately, it depends on feedback from the world to know if something new actually works or not. Predicting the weather five days into the future requires that you wait five days to find out if it worked. Although old data may provide a good source for learning how to predict the weather, a

forecast is only valuable if it tells us what the future weather will actually be. Faster computers cannot make the weather appear any faster, and so the speed at which a system can improve itself is limited by the speed at which data appear, not just the speed of its computations.

Even if we solve all of the problems associated with general intelligence learning, the rate at which it can evolve its capabilities is limited by the speed with which the world can provide feedback, and that is not affected by computer processing capacity. It has taken us 50,000 years to invent the current state of intelligence, there is no telling how long it would take to invent our way to general intelligence and then to superintelligence.

Resources

Aubert, M., Setiawan, P., Oktaviana, A. A., Brumm, A., Sulistyarto, P. H., Saptomo, E. W., . . . Brand, H. E. A. (2018). Paleolithic cave art in Borneo. *Nature, 564,* 254–257. https://www.nature.com/articles/s41586-018-0679-9.epdf

Bayern, A. M. P. von, Danel, S., Auersperg, A. M. I., Mioduszewska, B., & Kacelnik, A. (2018). Compound tool construction by New Caledonian crows. *Scientific Reports, 8,* 15676.

Beran, M. J., Rumbaugh, D. M., & Savage-Rumbaugh, E. S. (1998). Chimpanzee (*Pan troglodytes*) counting in a computerized testing paradigm. *The Psychological Record, 48*(1), 3–20. http://opensiuc.lib.siu.edu/tpr/vol48/iss1/1

Bostrom, N. (2014). *Superintelligence: Paths, dangers, strategies*. Oxford, UK: Oxford University Press.

Boysen, S. T., Berntson, G. G., Shreyer, T. A., & Hannan, M. (1995). Indicating acts during counting by a chimpanzee (*Pan troglodytes*). *Journal of Comparative Psychology, 109,* 47–51.

Calvin, W. (2004). *A brief history of the mind*. Oxford, UK: Oxford University Press.

Clark, A. (1998). *Magic words: How language augments human computation*. doi:10.1017/CBO9780511597909.011; http://www.nyu.edu/gsas/dept/philo/courses/concepts/magicwords.html

Gabora, L. (2007). Mind. In R. A. Bentley, H. D. G. Maschner, & C. Chippendale (Eds.), *Handbook of theories and methods in archaeology* (pp. 283–296). Walnut Creek, CA: Altamira Press.

Good, I. J. (1965). Speculations concerning the first ultraintelligent machine. In F. Alt & M. Rubinoff (Eds.), *Advances in computers* (Vol. 6, pp. 31–88). New York: Academic

Press. https://vtechworks.lib.vt.edu/bitstream/handle/10919/89424/TechReport05-3.pdf?sequence=1

Hofstadter, D. R. (1999) [1979], *Gödel, Escher, Bach: An Eternal Golden Braid*. New York: Basic Books.

Kaminski, J., Tempelmann, S., Call, J., & Tomasello, M. (2009). Domestic dogs comprehend communication with iconic signs. *Developmental Science, 12,* 831–837. https://www.eva.mpg.de/psycho/pdf/Publications_2009_PDF/Kaminski_Tempelmann_Call_Tomasello_2009.pdf

MacPherson, K., & Roberts, W. A. (2013). Can dogs count? *Learning and Motivation, 44,* 241–251.

Markham, J. A., & Greenough, W. T. (2004). Experience-driven brain plasticity: Beyond the synapse. *Neuron Glia Biology, 1,* 351–363. doi:10.1017/s1740925x05000219; https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1550735

McCarthy, J., Minsky, M., Rochester, N., & Shannon, C. E. (1955). *A proposal for the Dartmouth Summer Research Project on Artificial Intelligence.* http://jmc.stanford.edu/articles/dartmouth/dartmouth.pdf

Neisser, U., Boodoo, G., Bouchard, T. J. J., Boykin, A. W., Brody, N., Ceci, S. J., . . . Urbina, S. (1996). Intelligence: Knowns and unknowns. *American Psychologist, 51,* 77–101. http://differentialclub.wdfiles.com/local--files/definitions-structure-and-measurement/Intelligence-Knowns-and-unknowns.pdf

Newell, A., Shaw, J. C., & Simon, H. A. (1958). Elements of a theory of human problem solving. *Psychological Review, 65,* 151–166.

Owano, N. (2013). Scotland lunar-calendar find sparks Stone Age rethink. Phys.org. https://phys.org/news/2013-07-scotland-lunar-calendar-stone-age-rethink.html

Pásztor, E. (2011). Prehistoric astronomers? Ancient knowledge created by modern myth. *Journal of Cosmology, 14.* http://journalofcosmology.com/Consciousness159.html

Pearl, J., & Hartnett, K. (2018). To build truly intelligent machines, teach them cause and effect. *Quanta Magazine.* https://www.quantamagazine.org/to-build-truly-intelligent-machines-teach-them-cause-and-effect-20180515

Pearl, J., & Mackenzie, D. (2018). *The book of why: The new science of cause and effect.* New York: Basic Books.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., . . . Hassabis, D. (2016). Mastering the game of go with deep neural networks and tree search. *Nature, 529,* 484–489. http://airesearch.com/wp-content/uploads/2016/01/deepmind-mastering-go.pdf

Simon, H. A., & Newell, A. (1971). Human problem solving: The state of the theory in 1970. *American Psychologist, 26,* 145–159. https://pdfs.semanticscholar.org/18ce/82b07ac84aaf30b502c93076cec2accbfcaa.pdf

Smithsonian National Museum of Natural History. (2016). Human characteristics: Brains: Bigger brains: Complex brains for a complex world. http://humanorigins.si.edu/human-characteristics/brains

Stern, H., & Davidson, N. E. (2015). Trends in the skill of weather prediction at lead times of 1–14 days. *Quarterly Journal of the Royal Meteorological Society, Part A, 141,* 2726–2736.

Sternberg, R. J., & Detterman, D. K. (Eds.). (1986). *What is intelligence?* Norwood, NJ: Ablex.

2 Human Intelligence

In this chapter we consider just what it means for a human to be intelligent. Computers do not have to solve the same problems in precisely the same way, but it is still necessary to understand just what problems human intelligence does solve. General intelligence must still solve the same range of problems that a human can solve, so understanding that range is a critical step in creating general intelligence.

Human intelligence is our best known example of an intelligent system. In the early days of computational intelligence, following the 1956 Dartmouth workshop, the goal was to describe every aspect of human intelligence with enough precision that it could be simulated on a machine. Since that time, many working in the field have found that practical applications of computational intelligence do not need to duplicate how people solve problems, but rather these workers have found ways to reduce the complexity of an intelligence task to something that can be accomplished by a computer. General intelligence, on the other hand, does not seem to be solvable in the same reductionist way. General intelligence may actually gain from a deeper understanding of the best example we have of general intelligence—us.

As discussed in the introduction, conceptions of human intelligence focus on tasks that we associate with higher cognitive functioning—the kind of tasks that the people whom we admire for their superior intelligence perform that we cannot. The ability to do work in the field of theoretical physics, the ability to compose great music, and the ability to play chess are among these. These characteristics involve tasks that have been invented by people over time, and they are tasks that usually require formal education.

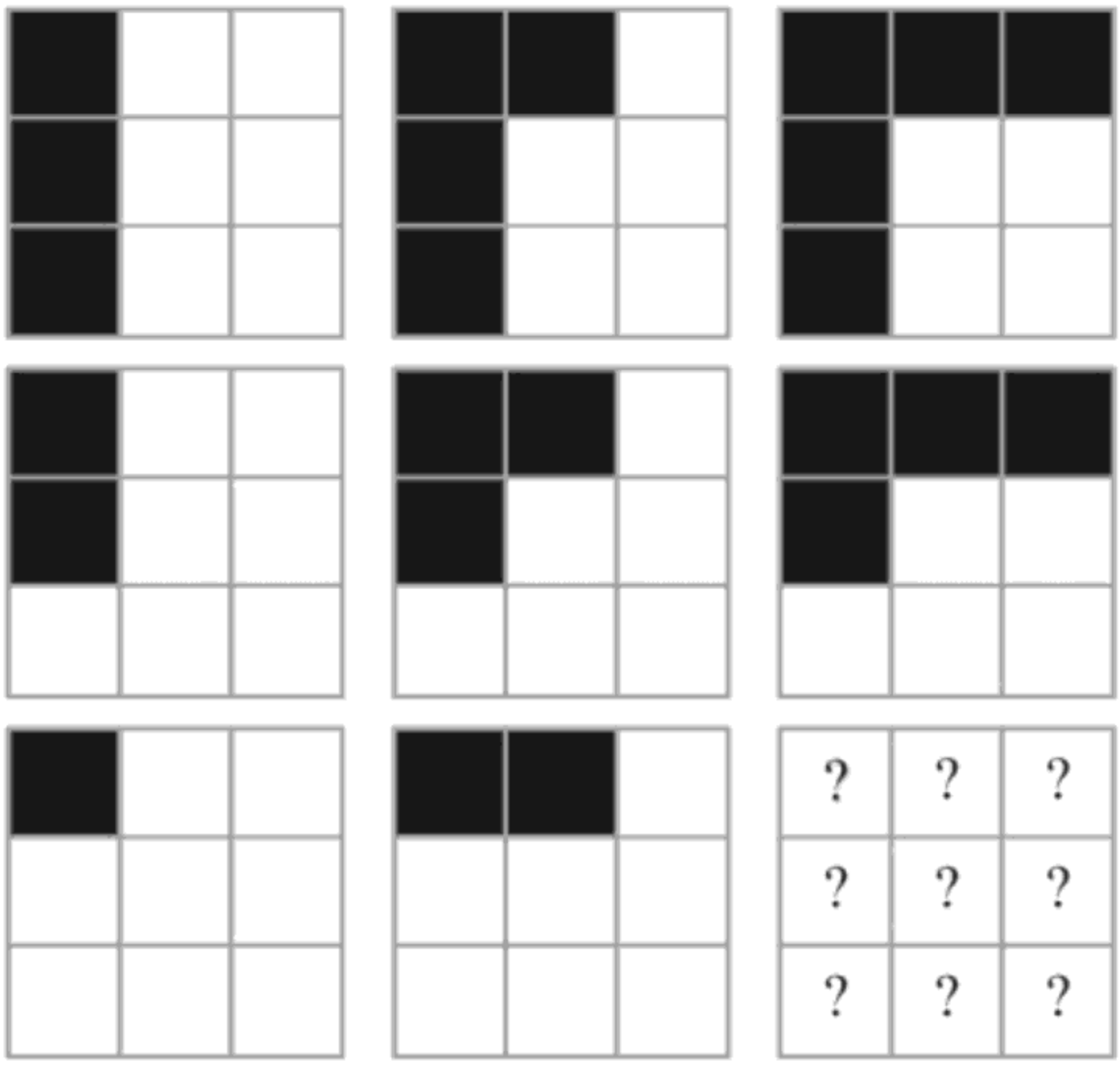

Figure 1
A simple example of a progressive matrix task used to assess intelligence. What pattern should be drawn in the ninth box that would be consistent with the previous squares in the row and in the column?

presented set of numbers (for example, what number would follow the sequence 2, 4, 6, 8?). In a progressive matrices task (see figure 1), the student is shown a matrix of designs exhibiting a certain pattern and must draw or choose the final design in that sequence. Both tasks require the student to induce the rule for the respective pattern and apply that rule. They both, in other words, tap some overlapping set of skills, and this overlap could be the cause of the correlation.

The jury is still out on whether there is such a *thing* in humans as general intelligence, at least as measured by intelligence tests. Computer scientists and psychologists have both been searching for it, but it has so far proven to be elusive.

Intelligence, as measured by intelligence tests, has been found to correlate with many intellectual capabilities, but not always the ones you might expect. It seems, for example, to have a weak relationship, if any, to complex problem-solving ability (Wenke, Frensch, & Funke, 2005).

Problem Solving

The ability to solve problems is a common feature among definitions of intelligence. Fortunately, this capability has also been well studied by psychologists and may provide an alternative means to get at the nature of intelligence.

Well-Formed Problems

In order for testers to be able to score intelligence tests, the tests must consist of specific questions that have specific answers. Real-world problems, on the other hand, often involve a large number of potential variables in complex relations. The goals of real-world problems may be unclear, and a substantial part of solving them is just finding the right goals. Studies of human problem solving involve well-formed problems because they are easy to administer, easy to score, and relatively easy to understand.

These laboratory tasks involve well-understood problems, and their outcomes are easy to evaluate. Games like chess, and now go, are complex, but they are very well-defined by their rules and by the position of the pieces during the game. There may be a lot of potential moves, but all of the valid moves are easy to identify.

Although there are laboratory studies of how people play chess, many psychological studies of problem solving have focused on simpler well-formed problems to be able to examine the entire problem-solving process in a reasonable amount of time. Three of these are the 8-tile problem, the Towers of Hanoi problem, and the hobbits and orcs problem (all three problems will be described shortly). These are simple enough to be solved in a brief laboratory session; the state of the problem is easy to describe without uncertainty. Finally, they do not rely on any particular knowledge to be able to solve them.

The 8-tile problem consists of a square frame containing eight tiles, numbered 1 through 8, and one empty spot. The digits are originally in some random order, and the solver's task is to arrange them in numerical order. The initial order is the "starting state," and the correct numerical order is the "goal state." Each step in solving the problem consists of moving one of the tiles into the empty slot. Only one tile can be moved at a time, and only a tile adjacent to the empty slot can be moved. Given a starting position, we could exhaustively list the succession of possible moves. We could even

draw a diagram of those possible moves. Each specific arrangement of the tiles is a "state" and the set of all possible arrangements is the "state space" for the problem. As in chess, the problem can be represented as a tree (see chapter 1), where each choice is a branch of the tree.

We solve the problem by successfully moving through this state space from the starting position through some sequence of selected states (by moving a tile) and finally reaching the goal state. We could choose a path through the state space by selecting the move at each point that gets us closer to the goal state.

Here is an example of one starting configuration. The empty tile is in the middle row and middle column:

1	4	3
7		6
5	8	2

From this configuration, there are four possible moves. We could move either the 4-tile, the 6-tile, the 7-tile, or the 8-tile into the blank space because these numbers are adjacent to the empty space. If the 4-tile is chosen, then the empty space will be in the center of the top row, as shown by the next configuration:

1		3
7	4	6
5	8	2

Then, on the next step, either the 1- or 3- or 4-tile could be moved, and so on.

The second commonly studied problem is the so-called Towers of Hanoi problem. See figure 2.

The puzzle was first described by Eduardo Lucas in 1883. In Lucas's version, the towers were supposed to be in an Indian temple dedicated to Brahma. In the more commonly known version, described by Sam Loyd

Figure 2
The three disk version of the Towers of Hanoi problem. The goal is to move the three disks from the first spindle to the last spindle following the rules of the task.

(1914), it was described as a problem being solved by monks in a fictitious temple in Hanoi, Vietnam. Supposedly, in the temple, the monks have to move a stack of 365 disks from one spindle to another. In the laboratory version, only three disks are typically used.

The laboratory version consists of three spindles and three disks of varying sizes. The starting state has the three disks stacked onto spindle 1 with the largest disk on the bottom and the smallest disk on the top. The puzzle solver's job is to move the disks from the first spindle to the third one, while obeying certain rules. Only one disk can be moved at a time, only one disk can be off of a spindle at a time, and a larger disk can never be placed on top of a smaller one (see, for example, Anzai & Simon, 1979, who studied solving a five-disk, three-spindle version of this problem).

With three disks and three spindles, there are only a few possible states. Initially, all three disks are on the first spindle. With three disks, the problem can be solved in a minimum of seven moves:

1. Move the smallest disk to the third spindle.
2. Move the medium disk to the middle spindle.
3. Move the small disk to the middle spindle.
4. Move the large disk to the third spindle.
5. Move the small disk to the first spindle.
6. Move the medium disk to the third spindle.
7. Move the small disk to the third spindle, and we are done.

As with the 8-tile problem, the number of states with three disks can be listed out explicitly. The problem is small enough to be solved in a short laboratory session. As the number of disks increases, though, the minimum number of moves needed to solve it grows exponentially. With 64 disks, and a move every second, it would take 585 billion years to solve.

The number of moves essentially doubles with each additional disk. Even though solving the puzzle with a large number of disks would take a very long time, the rules for solving it are easy to describe.

In the hobbits and orcs problem, three hobbits and three orcs arrive at a riverbank, and they all wish to cross to the other side (see Jeffries, Polson, Razran, & Atwood, 1977). There is a boat, but it can hold only two creatures at a time (two hobbits, two orcs, or one of each). If the orcs on one side of the river outnumber the hobbits, they will eat the hobbits, so you must be sure that there are never more orcs than hobbits on either side of the river. Other than the orcs' uncontrollable appetite for hobbits, all six of the creatures arriving at the river can otherwise be trusted. How can you get the six creatures across without losing any hobbits?

Here is a solution to this problem. "H" represents a hobbit. "O" represents an orc. The arrangement of hobbits and orcs on each side of the river constitutes the state of the problem, and the boat represents the transitions between states. See table 1.

These three simple problems, like the more complex ones such as go, chess, or checkers, are called "path problems." They can be described by a set of states and a set of actions (called "operators") for moving from one

Table 1

Description	Left Bank	Right Bank
All six arrive at the river	OOO HHH	
Send 2 orcs across	O HHH	OO
1 orc returns with the boat	OO HHH	O
Send 2 orcs across	HHH	OOO
1 orc returns with the boat	HHH O	OO
Send 2 hobbits across	O H	OO HH
1 hobbit and 1 orc return with the boat	OO HH	O H
Send 2 hobbits across	OO	O HHH
1 orc returns with the boat	OOO	HHH
Send 2 orcs across	O	OO HHH
1 orc returns with the boat	OO	O HHH
Send 2 orcs across		OOO HHH
Problem solved		Goal state

Premise: Bossy is a cow.

Premise: All cows are mortal.

Conclusion: Therefore Bossy is mortal.

Newell and Simon's General Problem Solver was a formal system in that it consisted of a set of basic tokens (axioms) and rules to manipulate them to make inferences. Games like checkers, chess, or go are formal systems because they consist of the basic pieces (the board and the playing pieces) and rules by which they can be manipulated. The pieces may have some meaning (for example, the knight and the bishop of chess), but one could effectively play chess without knowing their meaning, or even without any physical pieces at all.

The board and the positions of the chess pieces can be represented symbolically. For example, on one notation, each square on the chessboard is represented by a letter, indicating the square's column, and a number, indicating the square's row, similar to how we denote the cells in a spreadsheet. Each piece is represented by an uppercase letter, for example, Q for queen, R for rook (castle). A move is expressed by the symbol for the piece and the coordinate to which it is moved. The move Be5 means to move a bishop to the square e5. The whole game can be conducted using this symbolic notation or some other notation without ever touching physical pieces or a physical board.

Although formal reasoning is very important to intelligence, it is not all there is. In the next chapter, we will take up this question from a computational perspective. From a human cognition point of view, however, the evidence is clear that people do not inherently think logically. Logical thinking takes special effort.

Intelligence and formal reasoning imply rational decision-making. They imply that the reasoner will choose operators that advance it toward the goal. In general, a rational decision is one that is based on objective facts and that maximizes a desired benefit. Unless we are willing to just make up willy-nilly goals that fit whatever a person does, human decision-making often fails to be rational. Some people smoke, even though they know that there are health risks involved. We can imagine that there must be some goal that is rationally furthered by smoking, but that is circular reasoning. It makes up the goal to match the action and then tries to explain the action by this made-up goal. There may be some goal that is rationally

furthered by jumping out of perfectly good airplanes or by leaping on a grenade to save one's comrades. That last one may be heroic, but it is not in the personal interest of the hero to do it—it appears to be irrational.

Rational decisions are based on solid evidence and statistics. Rational decisions are often the more intelligent choice. People who make better, more rational decisions are usually perceived as being more intelligent than those who do not. One of the roles that logic plays, for example, is to help people reason systematically about the choices that they make. If the form is right, then the right decision should be consistently reached if people were rational decision makers. But they are not, at least not always.

For example, Amos Tversky and Daniel Kahneman found a number of situations in which people fail to make rational decisions. For instance, they found that people make different decisions under formally identical situations depending on how that situation is described. An example of this is that when graduate students were told of a penalty for registering for a conference after a particular date, 93% of them registered early, that is, before that date. When offered an identical early registration discount (that is, one with the same price difference before versus after the date), only 67% of them registered before that date. The two situations are identical, with the same benefit for registering early. The only difference was the label given to the action (penalty versus discount), but this label made a substantial difference. The students sought to avoid a loss described as a penalty but did not go out of their way for a gain.

Historically, this difference would have been interpreted as evidence that emotion intruded on the decision-making process and led the students to make an emotional rather than a logical decision. There is another possibility, however, that suggests that this deviation from rational decision-making was not a failure, but evidence for other processes that may play a role in intelligence. In fact, a formal system cannot be sufficient, even for logical reasoning.

A formal system depends solely on its internal structure, but intelligence requires interaction with a world, a world that includes uncertainty. A formal system starts with a set of basic premises, assumptions, or axioms. If the axioms are true, and the statements are of the right form, then the conclusions must also be true. The formal system assumes that the axioms are true, but there is no guarantee that this assumption is correct. The formal

system depends on the truth of the axioms, but by itself, it cannot establish their truth.

In logic, the axioms are typically called "premises." The premises could be wrong. For example, in the cow syllogism, we could assume that Bossy is a cow. We could further assume that all cows are mortal. Using the rules of the system, we could then infer that therefore Bossy is mortal. So far, so good, but how do we know that Bossy is actually a cow? That assumption could be wrong and there is no formal method to prove that it is true. If the axioms are not true, then any conclusions derived from those faulty axioms may also be faulty. If Bossy only looked like a cow but was actually an advanced robot, she might not, in fact, be mortal.

We might do tests to show that Bossy is a cow. But no matter how many tests we did, and no matter how many she passed, there is still a chance that we could be wrong, that the very next test we ran would indicate that she is a robot and not a cow.

We cannot prove that an axiom or premise is actually true. Deductions can be proved from the premises, but the premises cannot. We cannot infallibly move from specific observations to general truths. That inference must transcend logic. It depends critically on real-world facts, and there is no formal system that can prove that those facts are correct.

Starting in the late 1920s, a group of philosophers tried to create an approach to science that was strictly logical. In their view, scientists were misled when Newtonian mechanics was "replaced" by quantum mechanics. The basic principles of physics were not as Newton had described them. The logical positivists, as this group was known, tried to reduce science to just observation statements and logical deductions from those observations. If they could eliminate the sloppy language that was inherent in scientific theories, they argued, science would never be deceived again.

Observation statements (like "The temperature of the mixture increased by 2 degrees"), they thought, could be infallible as long as they were made with a healthy mind, that is, they ruled out hallucinations and the like as valid observation statements.

Without getting too far into the philosophical details, the approach of logical positivism failed. No purely logical system could produce science. Observations could be mistaken. Not every scientific statement could be immediately verified. As Kurt Gödel showed, not even mathematics, the most systematic and logical approach to knowledge that there is, could

survive as a complete system based solely on observation statements and logical deductions from them. Thomas Kuhn and later Imre Lakatos countered the logical positivists with a more psychological approach to scientific thinking.

Therefore, if the two examples that were arguably the most typical of human intelligence could not survive based on pure logic, it is extremely unlikely that similar processes could be the sole cause of human intelligence. Human intelligence has to go beyond mere observation and deductions from those observations.

Establishing the truth of a premise requires an inference. Inferences are always subject to uncertainty. We might think that we are playing a game of chess, but if, in fact, it only looked like chess, then the formal properties of the game might be different and success of the formal system would fly out the window.

Much of the science of computer science derives from treating computer algorithms as formal systems that can be proven to be true. An algorithm does not care what the computations represent, only that it is in the right form, and, if it is in the right form, it can be proved to be correct. The meaning of the variables in an algorithm does not affect the validity of the process. Two plus two equals four whether it is two ducks, two trucks, or two bucks. Algorithms do not care what they are reasoning about, but people often do.

Unlike formal systems, human intelligence often depends critically on the content of what we are thinking about. Humans are capable of believing things that are not true. Human language can express sentences that are neither true nor false, such as "This sentence is false." Humans interact with an uncertain world.

People have to go to school to learn logic, and many people find it difficult. If logic were the basis of human thought, then it would come "naturally," like walking. People who are educated to take advantage of formal systems are often able to accomplish tasks that they would not be able to do without such tools. On the other hand, simpler, more intuitive processes can often succeed where complicated formal systems would either take too long or be unduly affected by irrelevant information.

As discussed in chapter 1, people employ heuristics to guide much of their thought. A heuristic is a practical method that generally works, but, unlike an algorithm, is not guaranteed to produce the correct result. Typically, for

example, the taller child is likely to be the older child, but this heuristic can sometimes be wrong. One of the values of heuristics is that they allow people to reach conclusions that may not be fully justifiable but still may be valuable. The conclusion may not be provable, but it may take only a small amount of effort to reach it, and still be accurate enough for practical purposes. Because heuristics sometimes fail, they may also lead to false conclusions and prejudices that can sometimes interfere with intelligent action. They can have both value and cost and still contribute positively.

One heuristic that people use is called the "availability heuristic." People base their judgment on the examples that they can most easily bring to mind. Items that can be recalled most easily are treated as if they were the most representative examples and, therefore, the most important examples for making decisions.

The availability heuristic depends on unwarranted assumptions, but practically speaking, it can often be an effective way of dealing with real-world situations. Often the easiest to remember items are, in fact, the most relevant to the judgment. For example, if judging whether Chicago or Boston is the larger city, a full analysis might give a good answer, but availability might also provide an answer.

Under certain circumstances, the consequences of using the availability heuristic can sometimes conflict with a well-reasoned analysis, but under other circumstances, its use may be at least as accurate as a formal process. Unlike a detailed analysis, heuristic answers are often much faster and require much less effort than an exhaustive analysis.

If you were using availability to choose the larger city, you would decide that Chicago is the larger city if facts about it are more available than facts about Boston. If it is easier to call to mind facts about one city than another, the more fact-related city is likely to be the bigger one.

We cannot know directly how available memories are of these two cities for any specific person, but we can use another heuristic to estimate availability. We can, for example, look at the number of mentions each of these cities has in Google. The thinking is that if a city is mentioned more often in Google, then it is likely to be easier to think of the facts that are mentioned. This too, is a heuristic.

A Google search for "Chicago" at the end of 2019 claimed about 3 billion hits and a similar search for "Boston" claimed about 1.9 billion. Also according to Google, the population of Chicago is listed as 2.7 million, and

Another example of an insight problem is the socks problem. You are told that there are individual brown socks and black socks in a drawer in the ratio of five black socks for every four brown socks. How many socks do you have to pull out of the drawer to be certain to have at least one pair of either color? Drawing two socks is obviously not enough because they could be of different colors.

Many (educated) people approach this problem as a sampling question. They try to reason from the ratio of black to brown socks how big a sample they would need to be sure to get a complete pair. In reality, however, the ratio of sock colors is a distraction. No matter what the ratio, the correct answer is that you need to draw three socks to be sure to have a matched pair. Here's why:

With two colors, a draw of three socks is guaranteed to give you one of the following outcomes:

Black, black, black—pair of black socks

Black, black, brown—pair of black socks

Black, brown, brown—pair of brown socks

Brown, brown, brown—pair of brown socks

The ratio of black to brown socks can affect the relative likelihood of each of these four outcomes, but only these four are possible if three socks are selected. The selection does not even have to be random. Once we have the insight that there are only four possible outcomes, the problem's solution is easy.

Insight problems are typically posed in such a way that there are multiple ways that they could be represented. Archimedes was stymied as long as he thought about measuring the volume of the crown with a ruler or similar device. People solving the socks problem were stymied as long as they thought of the problem as one requiring the estimate of a probability. How you think about a problem, that is, how you represent what the problem is, can be critical to solving it.

Interesting insight problems typically require the use of a relatively uncommon representation. The socks problem is interesting because, for most people, the problem is most likely to evoke a representation centered on the ratio of 5:4, but this is a red herring. The main barrier to solving insight problems like this is to abandon the default representation and adopt a more productive one. Once the alternative representation is identified, the

rest of the problem-solving process may be very rapid. Laboratory versions of insight problems generally do not require any specific deep technical knowledge. Most of them can be solved by gaining one or two insights that change the nature of how the solver thinks about the problem.

Most of the problems given to computers for solution are well-structured path problems. The designer of the program provides the problem, its representation, and the operations that can move the computer toward its goal. It may be difficult to find a path to solution, using the representations, operators, and paths, because of the large number of possible states involved, but it is still a process of searching for and following a path. Insight problems, on the other hand, generally do not have a clear path. Computational intelligence research has not given serious attention to problems like these, but they are a clearly among the kinds of problems that an intelligent agent would have to address.

Here are a few more insight problems. The mutilated checkerboard was first described by Max Black in 1946. A regular checkerboard has 32 black squares and 32 red squares. If we had 32 dominoes, each the size of two squares, it would be obvious that we could cover the checkerboard with those 32 dominoes, for example, using 8 rows of 4 dominoes each. If we cut off the red square at the upper left corner of the checkerboard and the red square in the lower right corner of the checkerboard, could we now cover the mutilated checkerboard with 31 dominoes?

Another insight problem, the Königsberg bridges problem, is shown in figure 3. The city of Königsberg (now called Kaliningrad, Russia) was built on both sides of the Pregel River. Seven bridges connected two islands and the two sides of the river. Can you walk through the city, crossing the seven bridges each exactly once? In the map in figure 3, the bridges are marked in gray.

Here is a sequence of four numbers: 8, 5, 4, 9. Predict the next number in this sequence.

The two-strings problem was studied by Maier (1931). You are in a room with two strings hanging from the ceiling. Your task is to tie them together. In the room with you and the strings are a table, a wrench, a screwdriver, and a lighter. The strings are far enough apart that you cannot reach them both at the same time. How can these strings be tied together?

For the mutilated checkerboard problem we find that 8 rows of 4 dominoes will not work because two of the rows are short half a domino, but

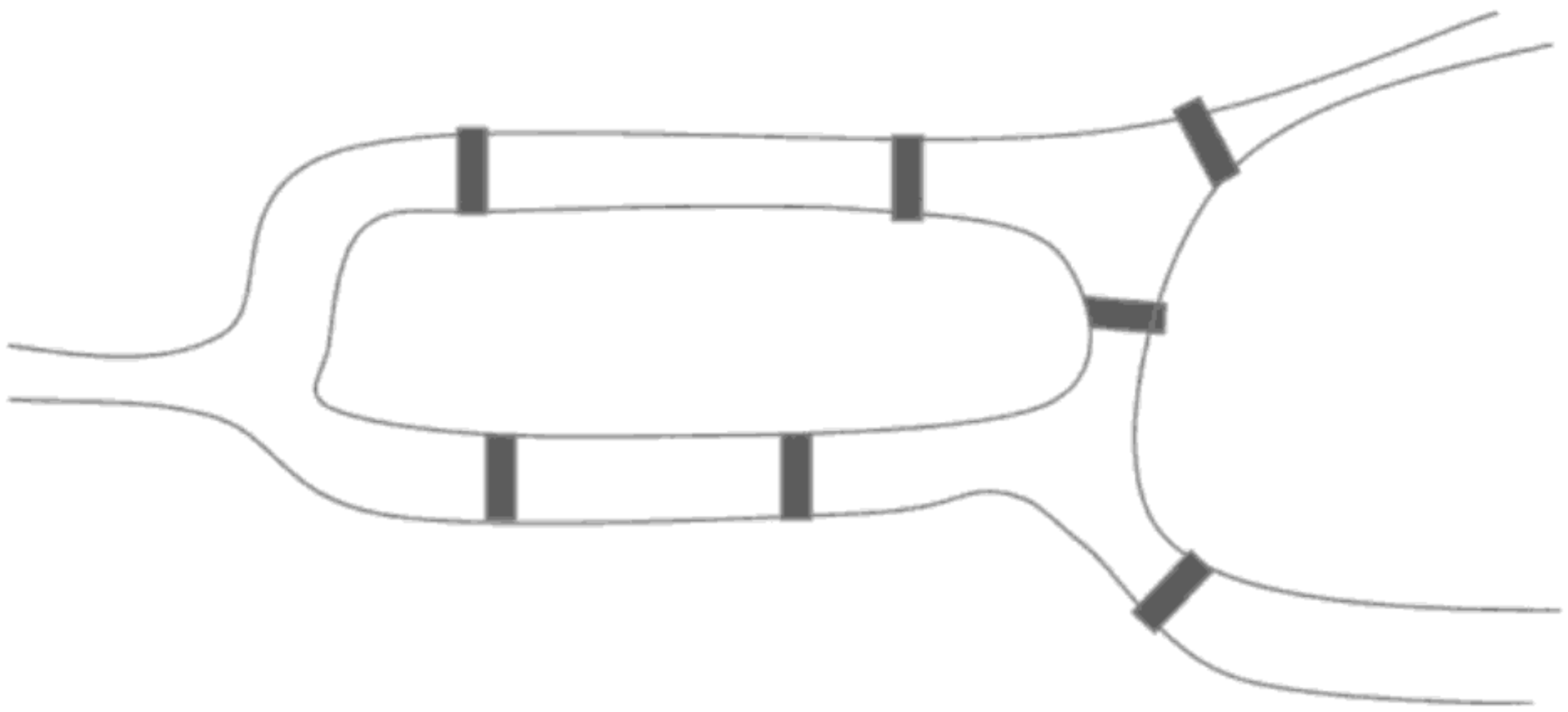

Figure 3
A sketch of the bridges connecting the land areas in Königsberg. Can you cross all seven bridges exactly once?

perhaps there is some arrangement of dominos that might work. You could try to lay out real or imaginary dominoes on the mutilated board, but when a particular pattern did not work, you would not know whether it was that pattern that was no good or whether there is no pattern that would work. Representing the problem in terms of dominoes and layouts makes solving the problem difficult at best. In theory, a computer could use this rearrangement method to try to determine whether the board can be covered by 31 dominoes, but it requires testing all possible arrangements. In the absence of insight, we have only brute force. There are no approximate solutions that can be used to help us search the tree of possible arrangements. We just have to try them.

Before we go back to the mutilated checkerboard problem, consider this one. There are 32 men and 32 women at a dance. Only heterosexual couples dance. Can everyone at the party dance at the same time? Now two of the women leave the party. Can we still form 31 heterosexual couples?

In the original checkerboard, each domino covered exactly one red square and one black square. Each heterosexual dance couple must contain exactly one man (black square) and one woman (red square). In the mutilated checkerboard, there are 32 black squares but only 30 red squares. Representing the problem this way reveals that it is impossible to cover a mutilated checkerboard exactly with 31 dominos even though there are exactly 62 squares. The mutilated checkerboard problem is formally identical to the

heterosexual dance problem. People tend to find the dance problem relatively easy but find the checkerboard problem relatively difficult.

The mutilated checkerboard problem can be solved using a brute-force solution where every layout of the dominoes is tried. Trying a few thousand potential layouts may be practical with an 8×8 board but may not be practical with a much larger analogous board. There are 6,728 ways to arrange dominoes on a regular 8×8 checkerboard. But if we increase the number of squares to form a 12×12 "checkerboard," the number of possible domino arrangements grows to 53,060,477,521,960,000. With the insight that a domino must cover exactly one red and one black square, on the other hand, we can instantly solve the problem no matter how many squares are on the board.

An expert might recognize the mutilated checkerboard and the dance party problem as examples of a parity problem and solve both of them even more quickly. The dance party problem is much easier to solve because the useful representation is much more obvious, meaning that people are likely to come up with it quickly. Solving the dance problem can help solve the checkerboard problem if you can see the relationship between the two problems. Current approaches to computational intelligence generally cannot take advantage of this analogy. To be fair, many people fail to see the connection as well (Gick & McGarry, 1992).

The Königsberg bridges problem is also similar. Königsberg is divided into four regions. Each bridge connects exactly two regions. Except at the start or the end of the walk, every time one enters a region by a bridge, one must leave the region by a bridge. The number of times one enters must equal the number of times one leaves it, so the number of bridges touching a land mass must be an even number to cross them all exactly once because half of them will be used to enter a region and half will be used to leave it. The only possible exceptions are the regions where you start your walk and where you end it. Only a city with exactly none or exactly two regions with an odd number of bridges (one where you start and one where you finish) can be walked without repetition. In Königsberg, each region is served by an odd number of bridges, so there is no way that one can walk the seven bridges exactly once.

The checkerboard, dance, and bridges problems are related. They can all be represented as graphs (nodes connected by arcs). For our purposes, these three problems illustrate two things. How you represent the problem can