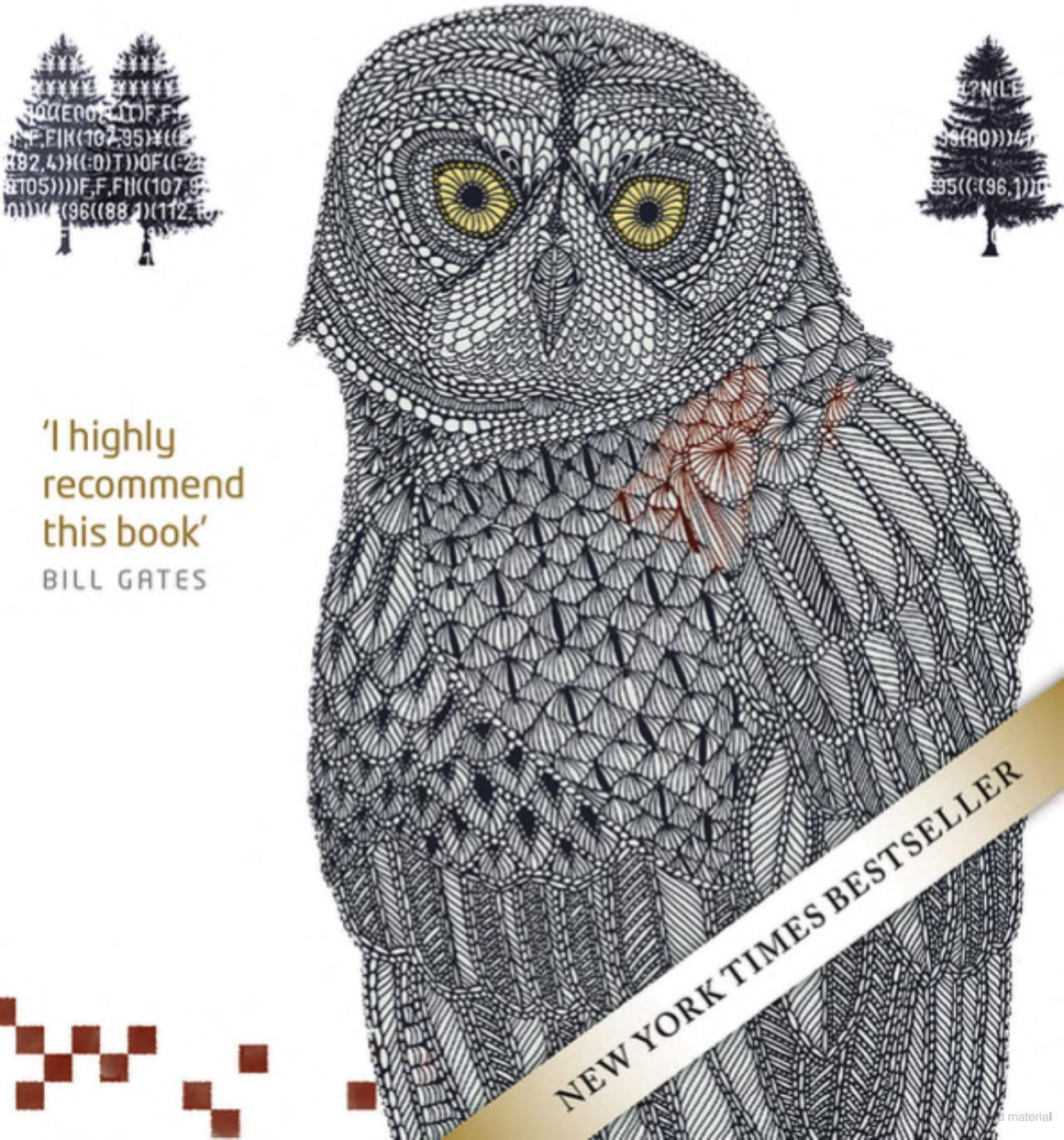
NEW AFTERWORD

NICK BOSTROM

SUPERINTELLIGENCE

Paths, Dangers, Strategies



'I highly recommend this book'
BILL GATES

NEW YORK TIMES BESTSELLER

SUPERINTELLIGENCE

Paths, Dangers, Strategies

NICK BOSTROM

Director, Future of Humanity Institute
Professor, Faculty of Philosophy & Oxford Martin School
University of Oxford

OXFORD
UNIVERSITY PRESS

OXFORD
UNIVERSITY PRESS

Great Clarendon Street, Oxford, OX2 6DP,
United Kingdom

Oxford University Press is a department of the University of Oxford.
It furthers the University's objective of excellence in research, scholarship,
and education by publishing worldwide. Oxford is a registered trade mark of
Oxford University Press in the UK and in certain other countries

First Edition published in 2014
Reprinted with corrections 2017

Impression: 17

British Library Cataloguing in Publication Data
Data available

Library of Congress Control Number: 2013955152

ISBN 978–0–19–967811–2

Printed in Great Britain by
Clays Ltd, St Ives plc

CONTENTS

LISTS OF FIGURES, TABLES, AND BOXES

List of Figures

List of Tables

List of Boxes

CHAPTER 1

Past developments and present capabilities

We begin by looking back. History, at the largest scale, seems to exhibit a sequence of distinct growth modes, each much more rapid than its predecessor. This pattern has been taken to suggest that another (even faster) growth mode might be possible. However, we do not place much weight on this observation—this is not a book about "technological acceleration" or "exponential growth" or the miscellaneous notions sometimes gathered under the rubric of "the singularity." Next, we review the history of artificial intelligence. We then survey the field's current capabilities. Finally, we glance at some recent expert opinion surveys, and contemplate our ignorance about the timeline of future advances.

Growth modes and big history

A mere few million years ago our ancestors were still swinging from the branches in the African canopy. On a geological or even evolutionary timescale, the rise of *Homo sapiens* from our last common ancestor with the great apes happened swiftly. We developed upright posture, opposable thumbs, and—crucially—some relatively minor changes in brain size and neurological organization that led to a great leap in cognitive ability. As a consequence, humans can think abstractly, communicate complex thoughts, and culturally accumulate information over the generations far better than any other species on the planet.

These capabilities let humans develop increasingly efficient productive technologies, making it possible for our ancestors to migrate far away from the rainforest and the savanna. Especially after the adoption of agriculture, population densities rose along with the total size of the human population. More people meant more ideas; greater densities meant that ideas could spread more readily and that some individuals could devote themselves to developing specialized skills. These

developments increased the *rate of growth* of economic productivity and technological capacity. Later developments, related to the Industrial Revolution, brought about a second, comparable step change in the rate of growth.

Such changes in the rate of growth have important consequences. A few hundred thousand years ago, in early human (or hominid) prehistory, growth was so slow that it took on the order of one million years for human productive capacity to increase sufficiently to sustain an additional one million individuals living at subsistence level. By 5000 BC, following the Agricultural Revolution, the rate of growth had increased to the point where the same amount of growth took just two centuries. Today, following the Industrial Revolution, the world economy grows on average by that amount every ninety minutes.[1]

Even the present rate of growth will produce impressive results if maintained for a moderately long time. If the world economy continues to grow at the same pace as it has over the past fifty years, then the world will be some 4.8 times richer by 2050 and about 34 times richer by 2100 than it is today.[2]

Yet the prospect of continuing on a steady exponential growth path pales in comparison to what would happen if the world were to experience another step change in the *rate of growth* comparable in magnitude to those associated with the Agricultural Revolution and the Industrial Revolution. The economist Robin Hanson estimates, based on historical economic and population data, a characteristic world economy doubling time for Pleistocene hunter–gatherer society of 224,000 years; for farming society, 909 years; and for industrial society, 6.3 years.[3] (In Hanson's model, the present epoch is a mixture of the farming and the industrial growth modes—the world economy as a whole is not yet growing at the 6.3-year doubling rate.) If another such transition to a different growth mode were to occur, and it were of similar magnitude to the previous two, it would result in a new growth regime in which the world economy would double in size about every two weeks.

Such a growth rate seems fantastic by current lights. Observers in earlier epochs might have found it equally preposterous to suppose that the world economy would one day be doubling several times within a single lifespan. Yet that is the extraordinary condition we now take to be ordinary.

The idea of a coming technological singularity has by now been widely popularized, starting with Vernor Vinge's seminal essay and continuing with the writings of Ray Kurzweil and others.[4] The term "singularity," however, has been used confusedly in many disparate senses and has accreted an unholy (yet almost millenarian) aura of techno-utopian connotations.[5] Since most of these meanings and connotations are irrelevant to our argument, we can gain clarity by dispensing with the "singularity" word in favor of more precise terminology.

The singularity-related idea that interests us here is the possibility of an *intelligence explosion*, particularly the prospect of machine superintelligence. There may be those who are persuaded by growth diagrams like the ones in Figure 1 that another drastic change in growth mode is in the cards, comparable to the Agricultural or Industrial Revolution. These folk may then reflect that it is hard

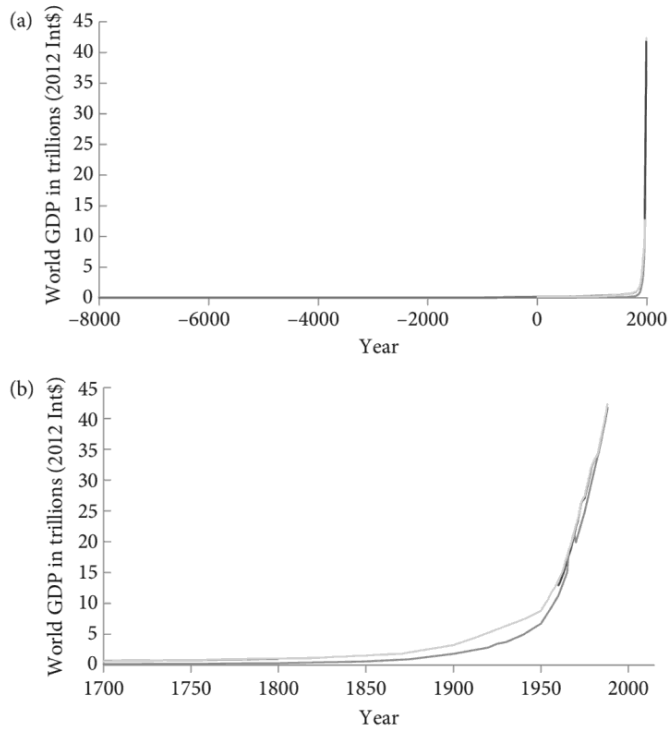


Figure 1 Long-term history of world GDP. Plotted on a linear scale, the history of the world economy looks like a flat line hugging the *x*-axis, until it suddenly spikes vertically upward. (a) Even when we zoom in on the most recent 10,000 years, the pattern remains essentially one of a single 90° angle. (b) Only within the past 100 years or so does the curve lift perceptibly above the zero-level. (The different lines in the plot correspond to different data sets, which yield slightly different estimates.[6])

to conceive of a scenario in which the world economy's doubling time shortens to mere weeks that does not involve the creation of minds that are much faster and more efficient than the familiar biological kind. However, the case for taking seriously the prospect of a machine intelligence revolution need not rely on curve-fitting exercises or extrapolations from past economic growth. As we shall see, there are stronger reasons for taking heed.

Great expectations

Machines matching humans in general intelligence—that is, possessing common sense and an effective ability to learn, reason, and plan to meet complex information-processing challenges across a wide range of natural and abstract domains—have been expected since the invention of computers in the 1940s. At that time, the advent of such machines was often placed some twenty years into

the future.[7] Since then, the expected arrival date has been receding at a rate of one year per year; so that today, futurists who concern themselves with the possibility of artificial general intelligence still often believe that intelligent machines are a couple of decades away.[8]

Two decades is a sweet spot for prognosticators of radical change: near enough to be attention-grabbing and relevant, yet far enough to make it possible to suppose that a string of breakthroughs, currently only vaguely imaginable, might by then have occurred. Contrast this with shorter timescales: most technologies that will have a big impact on the world in five or ten years from now are already in limited use, while technologies that will reshape the world in less than fifteen years probably exist as laboratory prototypes. Twenty years may also be close to the typical duration remaining of a forecaster's career, bounding the reputational risk of a bold prediction.

From the fact that some individuals have overpredicted artificial intelligence in the past, however, it does not follow that AI is impossible or will never be developed.[9] The main reason why progress has been slower than expected is that the technical difficulties of constructing intelligent machines have proved greater than the pioneers foresaw. But this leaves open just how great those difficulties are and how far we now are from overcoming them. Sometimes a problem that initially looks hopelessly complicated turns out to have a surprisingly simple solution (though the reverse is probably more common).

In the next chapter, we will look at different paths that may lead to human-level machine intelligence. But let us note at the outset that however many stops there are between here and human-level machine intelligence, the latter is not the final destination. The next stop, just a short distance farther along the tracks, is superhuman-level machine intelligence. The train might not pause or even decelerate at Humanville Station. It is likely to swoosh right by.

The mathematician I. J. Good, who had served as chief statistician in Alan Turing's code-breaking team in World War II, might have been the first to enunciate the essential aspects of this scenario. In an oft-quoted passage from 1965, he wrote:

> Let an ultraintelligent machine be defined as a machine that can far surpass all the intellectual activities of any man however clever. Since the design of machines is one of these intellectual activities, an ultraintelligent machine could design even better machines; there would then unquestionably be an "intelligence explosion," and the intelligence of man would be left far behind. Thus the first ultraintelligent machine is the last invention that man need ever make, provided that the machine is docile enough to tell us how to keep it under control.[10]

It may seem obvious now that major existential risks would be associated with such an intelligence explosion, and that the prospect should therefore be examined with the utmost seriousness even if it were known (which it is not) to have but a moderately small probability of coming to pass. The pioneers of artificial intelligence, however, notwithstanding their belief in the imminence of human-level

AI, mostly did not contemplate the possibility of greater-than-human AI. It is as though their speculation muscle had so exhausted itself in conceiving the radical possibility of machines reaching human intelligence that it could not grasp the corollary—that machines would subsequently become superintelligent.

The AI pioneers for the most part did not countenance the possibility that their enterprise might involve risk.[11] They gave no lip service—let alone serious thought—to any safety concern or ethical qualm related to the creation of artificial minds and potential computer overlords: a lacuna that astonishes even against the background of the era's not-so-impressive standards of critical technology assessment.[12] We must hope that by the time the enterprise eventually does become feasible, we will have gained not only the technological proficiency to set off an intelligence explosion but also the higher level of mastery that may be necessary to make the detonation survivable.

But before we turn to what lies ahead, it will be useful to take a quick glance at the history of machine intelligence to date.

Seasons of hope and despair

In the summer of 1956 at Dartmouth College, ten scientists sharing an interest in neural nets, automata theory, and the study of intelligence convened for a six-week workshop. This Dartmouth Summer Project is often regarded as the cockcrow of artificial intelligence as a field of research. Many of the participants would later be recognized as founding figures. The optimistic outlook among the delegates is reflected in the proposal submitted to the Rockefeller Foundation, which provided funding for the event:

> We propose that a 2 month, 10 man study of artificial intelligence be carried out. . . . The study is to proceed on the basis of the conjecture that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it. An attempt will be made to find how to make machines that use language, form abstractions and concepts, solve kinds of problems now reserved for humans, and improve themselves. We think that a significant advance can be made in one or more of these problems if a carefully selected group of scientists work on it together for a summer.

In the six decades since this brash beginning, the field of artificial intelligence has been through periods of hype and high expectations alternating with periods of setback and disappointment.

The first period of excitement, which began with the Dartmouth meeting, was later described by John McCarthy (the event's main organizer) as the "Look, Ma, no hands!" era. During these early days, researchers built systems designed to refute claims of the form "No machine could ever do *X*!" Such skeptical claims were common at the time. To counter them, the AI researchers created small systems that achieved *X* in a "microworld" (a well-defined, limited domain that enabled

a pared-down version of the performance to be demonstrated), thus providing a proof of concept and showing that *X* could, in principle, be done by machine. One such early system, the Logic Theorist, was able to prove most of the theorems in the second chapter of Whitehead and Russell's *Principia Mathematica*, and even came up with one proof that was much more elegant than the original, thereby debunking the notion that machines could "only think numerically" and showing that machines were also able to do deduction and to invent logical proofs.[13] A follow-up program, the General Problem Solver, could in principle solve a wide range of formally specified problems.[14] Programs that could solve calculus problems typical of first-year college courses, visual analogy problems of the type that appear in some IQ tests, and simple verbal algebra problems were also written.[15] The Shakey robot (so named because of its tendency to tremble during operation) demonstrated how logical reasoning could be integrated with perception and used to plan and control physical activity.[16] The ELIZA program showed how a computer could impersonate a Rogerian psychotherapist.[17] In the mid-seventies, the program SHRDLU showed how a simulated robotic arm in a simulated world of geometric blocks could follow instructions and answer questions in English that were typed in by a user.[18] In later decades, systems would be created that demonstrated that machines could compose music in the style of various classical composers, outperform junior doctors in certain clinical diagnostic tasks, drive cars autonomously, and make patentable inventions.[19] There has even been an AI that cracked original jokes.[20] (Not that its level of humor was high—"What do you get when you cross an *optic* with a *mental object*? An *eye*-dea"—but children reportedly found its puns consistently entertaining.)

The methods that produced successes in the early demonstration systems often proved difficult to extend to a wider variety of problems or to harder problem instances. One reason for this is the "combinatorial explosion" of possibilities that must be explored by methods that rely on something like exhaustive search. Such methods work well for simple instances of a problem, but fail when things get a bit more complicated. For instance, to prove a theorem that has a 5-line long proof in a deduction system with one inference rule and 5 axioms, one could simply enumerate the 3,125 possible combinations and check each one to see if it delivers the intended conclusion. Exhaustive search would also work for 6- and 7-line proofs. But as the task becomes more difficult, the method of exhaustive search soon runs into trouble. Proving a theorem with a 50-line proof does not take ten times longer than proving a theorem that has a 5-line proof: rather, if one uses exhaustive search, it requires combing through $5^{50} \approx 8.9 \times 10^{34}$ possible sequences—which is computationally infeasible even with the fastest supercomputers.

To overcome the combinatorial explosion, one needs algorithms that exploit structure in the target domain and take advantage of prior knowledge by using heuristic search, planning, and flexible abstract representations—capabilities that were poorly developed in the early AI systems. The performance of these early systems also suffered because of poor methods for handling uncertainty, reliance on brittle and ungrounded symbolic representations, data scarcity, and

severe hardware limitations on memory capacity and processor speed. By the mid-1970s, there was a growing awareness of these problems. The realization that many AI projects could never make good on their initial promises led to the onset of the first "AI winter": a period of retrenchment, during which funding decreased and skepticism increased, and AI fell out of fashion.

A new springtime arrived in the early 1980s, when Japan launched its Fifth-Generation Computer Systems Project, a well-funded public–private partnership that aimed to leapfrog the state of the art by developing a massively parallel computing architecture that would serve as a platform for artificial intelligence. This occurred at peak fascination with the Japanese "post-war economic miracle," a period when Western government and business leaders anxiously sought to divine the formula behind Japan's economic success in hope of replicating the magic at home. When Japan decided to invest big in AI, several other countries followed suit.

The ensuing years saw a great proliferation of *expert systems*. Designed as support tools for decision makers, expert systems were rule-based programs that made simple inferences from a knowledge base of facts, which had been elicited from human domain experts and painstakingly hand-coded in a formal language. Hundreds of these expert systems were built. However, the smaller systems provided little benefit, and the larger ones proved expensive to develop, validate, and keep updated, and were generally cumbersome to use. It was impractical to acquire a standalone computer just for the sake of running one program. By the late 1980s, this growth season, too, had run its course.

The Fifth-Generation Project failed to meet its objectives, as did its counterparts in the United States and Europe. A second AI winter descended. At this point, a critic could justifiably bemoan "the history of artificial intelligence research to date, consisting always of very limited success in particular areas, followed immediately by failure to reach the broader goals at which these initial successes seem at first to hint."[21] Private investors began to shun any venture carrying the brand of "artificial intelligence." Even among academics and their funders, "AI" became an unwanted epithet.[22]

Technical work continued apace, however, and by the 1990s, the second AI winter gradually thawed. Optimism was rekindled by the introduction of new techniques, which seemed to offer alternatives to the traditional logicist paradigm (often referred to as "Good Old-Fashioned Artificial Intelligence," or "GOFAI" for short), which had focused on high-level symbol manipulation and which had reached its apogee in the expert systems of the 1980s. The newly popular techniques, which included neural networks and genetic algorithms, promised to overcome some of the shortcomings of the GOFAI approach, in particular the "brittleness" that characterized classical AI programs (which typically produced complete nonsense if the programmers made even a single slightly erroneous assumption). The new techniques boasted a more organic performance. For example, neural networks exhibited the property of "graceful degradation": a small amount of damage to a neural network typically resulted in a small

degradation of its performance, rather than a total crash. Even more importantly, neural networks could learn from experience, finding natural ways of generalizing from examples and finding hidden statistical patterns in their input.[23] This made the nets good at pattern recognition and classification problems. For example, by training a neural network on a data set of sonar signals, it could be taught to distinguish the acoustic profiles of submarines, mines, and sea life with better accuracy than human experts—and this could be done without anybody first having to figure out in advance exactly how the categories were to be defined or how different features were to be weighted.

While simple neural network models had been known since the late 1950s, the field enjoyed a renaissance after the introduction of the backpropagation algorithm, which made it possible to train multi-layered neural networks.[24] Such multilayered networks, which have one or more intermediary ("hidden") layers of neurons between the input and output layers, can learn a much wider range of functions than their simpler predecessors.[25] Combined with the increasingly powerful computers that were becoming available, these algorithmic improvements enabled engineers to build neural networks that were good enough to be practically useful in many applications.

The brain-like qualities of neural networks contrasted favorably with the rigidly logic-chopping but brittle performance of traditional rule-based GOFAI systems—enough so to inspire a new "-ism," *connectionism*, which emphasized the importance of massively parallel sub-symbolic processing. More than 150,000 academic papers have since been published on artificial neural networks, and they continue to be an important approach in machine learning.

Evolution-based methods, such as genetic algorithms and genetic programming, constitute another approach whose emergence helped end the second AI winter. It made perhaps a smaller academic impact than neural nets but was widely popularized. In evolutionary models, a population of candidate solutions (which can be data structures or programs) is maintained, and new candidate solutions are generated randomly by mutating or recombining variants in the existing population. Periodically, the population is pruned by applying a selection criterion (a fitness function) that allows only the better candidates to survive into the next generation. Iterated over thousands of generations, the average quality of the solutions in the candidate pool gradually increases. When it works, this kind of algorithm can produce efficient solutions to a very wide range of problems—solutions that may be strikingly novel and unintuitive, often looking more like natural structures than anything that a human engineer would design. And in principle, this can happen without much need for human input beyond the initial specification of the fitness function, which is often very simple. In practice, however, getting evolutionary methods to work well requires skill and ingenuity, particularly in devising a good representational format. Without an efficient way to encode candidate solutions (a genetic language that matches latent structure in the target domain), evolutionary search tends to meander endlessly in a vast search space or get stuck at a local optimum. Even if a good representational

format is found, evolution is computationally demanding and is often defeated by the combinatorial explosion.

Neural networks and genetic algorithms are examples of methods that stimulated excitement in the 1990s by appearing to offer alternatives to the stagnating GOFAI paradigm. But the intention here is not to sing the praises of these two methods or to elevate them above the many other techniques in machine learning. In fact, one of the major theoretical developments of the past twenty years has been a clearer realization of how superficially disparate techniques can be understood as special cases within a common mathematical framework. For example, many types of artificial neural network can be viewed as classifiers that perform a particular kind of statistical calculation (maximum likelihood estimation).[26] This perspective allows neural nets to be compared with a larger class of algorithms for learning classifiers from examples—"decision trees," "logistic regression models," "support vector machines," "naive Bayes," "*k*-nearest-neighbors regression," among others.[27] In a similar manner, genetic algorithms can be viewed as performing stochastic hill-climbing, which is again a subset of a wider class of algorithms for optimization. Each of these algorithms for building classifiers or for searching a solution space has its own profile of strengths and weaknesses which can be studied mathematically. Algorithms differ in their processor time and memory space requirements, which inductive biases they presuppose, the ease with which externally produced content can be incorporated, and how transparent their inner workings are to a human analyst.

Behind the razzle-dazzle of machine learning and creative problem-solving thus lies a set of mathematically well-specified tradeoffs. The ideal is that of the perfect Bayesian agent, one that makes probabilistically optimal use of available information. This ideal is unattainable because it is too computationally demanding to be implemented in any physical computer (see Box 1). Accordingly, one can view artificial intelligence as a quest to find shortcuts: ways of tractably approximating the Bayesian ideal by sacrificing some optimality or generality while preserving enough to get high performance in the actual domains of interest.

A reflection of this picture can be seen in the work done over the past couple of decades on probabilistic graphical models, such as Bayesian networks. Bayesian networks provide a concise way of representing probabilistic and conditional independence relations that hold in some particular domain. (Exploiting such independence relations is essential for overcoming the combinatorial explosion, which is as much of a problem for probabilistic inference as it is for logical deduction.) They also provide important insight into the concept of causality.[28]

One advantage of relating learning problems from specific domains to the general problem of Bayesian inference is that new algorithms that make Bayesian inference more efficient will then yield immediate improvements across many different areas. Advances in Monte Carlo approximation techniques, for example, are directly applied in computer vision, robotics, and computational genetics. Another advantage is that it lets researchers from different disciplines more

Box 1 An optimal Bayesian agent

An ideal Bayesian agent starts out with a "prior probability distribution," a function that assigns probabilities to each "possible world" (i.e. to each maximally specific way the world could turn out to be).[29] This prior incorporates an inductive bias such that simpler possible worlds are assigned higher probabilities. (One way to formally define the simplicity of a possible world is in terms of its "Kolmogorov complexity," a measure based on the length of the shortest computer program that generates a complete description of the world.[30]) The prior also incorporates any background knowledge that the programmers wish to give to the agent.

As the agent receives new information from its sensors, it updates its probability distribution by conditionalizing the distribution on the new information according to Bayes' theorem.[31] Conditionalization is the mathematical operation that sets the new probability of those worlds that are inconsistent with the information received to zero and renormalizes the probability distribution over the remaining possible worlds. The result is a "posterior probability distribution" (which the agent may use as its new prior in the next time step). As the agent makes observations, its probability mass thus gets concentrated on the shrinking set of possible worlds that remain consistent with the evidence; and among these possible worlds, simpler ones always have more probability.

Metaphorically, we can think of a probability as sand on a large sheet of paper. The paper is partitioned into areas of various sizes, each area corresponding to one possible world, with larger areas corresponding to simpler possible worlds. Imagine also a layer of sand of even thickness spread across the entire sheet: this is our prior probability distribution. Whenever an observation is made that rules out some possible worlds, we remove the sand from the corresponding areas of the paper and redistribute it evenly over the areas that remain in play. Thus, the total amount of sand on the sheet never changes, it just gets concentrated into fewer areas as observational evidence accumulates. This is a picture of learning in its purest form. (To calculate the probability of a *hypothesis*, we simply measure the amount of sand in all the areas that correspond to the possible worlds in which the hypothesis is true.)

So far, we have defined a learning rule. To get an agent, we also need a decision rule. To this end, we endow the agent with a "utility function" which assigns a number to each possible world. The number represents the desirability of that world according to the agent's basic preferences. Now, at each time step, the agent selects the action with the highest expected utility.[32] (To find the action with the highest expected utility, the agent could list all possible actions. It could then compute the conditional probability distribution given the action—the probability distribution that would result from conditionalizing its current probability distribution on the observation that the action had just been taken. Finally, it could calculate the expected value of the action as the sum of the value

Box 1 *Continued*

of each possible world multiplied by the conditional probability of that world given the action.[33])

The learning rule and the decision rule together define an "optimality notion" for an agent. (Essentially the same optimality notion has been broadly used in artificial intelligence, epistemology, philosophy of science, economics, and statistics.[34]) In reality, it is impossible to build such an agent because it is computationally intractable to perform the requisite calculations. Any attempt to do so succumbs to a combinatorial explosion just like the one described in our discussion of GOFAI. To see why this is so, consider one tiny subset of all possible worlds: those that consist of a single computer monitor floating in an endless vacuum. The monitor has 1,000 × 1,000 pixels, each of which is perpetually either on or off. Even this subset of possible worlds is enormously large: the $2^{(1,000 \times 1,000)}$ possible monitor states outnumber all the computations expected ever to take place in the observable universe. Thus, we could not even enumerate all the possible worlds in this tiny subset of all possible worlds, let alone perform more elaborate computations on each of them individually.

Optimality notions can be of theoretical interest even if they are physically unrealizable. They give us a standard by which to judge heuristic approximations, and sometimes we can reason about what an optimal agent would do in some special case. We will encounter some alternative optimality notions for artificial agents in Chapter 12.

easily pool their findings. Graphical models and Bayesian statistics have become a shared focus of research in many fields, including machine learning, statistical physics, bioinformatics, combinatorial optimization, and communication theory.[35] A fair amount of the recent progress in machine learning has resulted from incorporating formal results originally derived in other academic fields. (Machine learning applications have also benefitted enormously from faster computers and greater availability of large data sets.)

State of the art

Artificial intelligence already outperforms human intelligence in many domains. Table 1 surveys the state of game-playing computers, showing that AIs now beat human champions in a wide range of games.[36]

These achievements might not seem impressive today. But this is because our standards for what is impressive keep adapting to the advances being made. Expert chess playing, for example, was once thought to epitomize human intellection. In the view of several experts in the late fifties: "If one could devise a successful chess

Table 1 ***Game-playing AI***

Checkers	Superhuman	Arthur Samuel's checkers program, originally written in 1952 and later improved (the 1955 version incorporating machine learning), becomes the first program to learn to play a game better than its creator.[37] In 1994, the program CHINOOK beats the reigning human champion, marking the first time a program wins an official world championship in a game of skill. In 2002, Jonathan Schaeffer and his team "solve" checkers, i.e. produce a program that always makes the best possible move (combining alpha-beta search with a database of 39 trillion endgame positions). Perfect play by both sides leads to a draw.[38]
Backgammon	Superhuman	1979: The backgammon program BKG by Hans Berliner defeats the world champion—the first computer program to defeat (in an exhibition match) a world champion in any game—though Berliner later attributes the win to luck with the dice rolls.[39]
		1992: The backgammon program TD-Gammon by Gerry Tesauro reaches championship-level ability, using temporal difference learning (a form of reinforcement learning) and repeated plays against itself to improve.[40]
		In the years since, backgammon programs have far surpassed the best human players.[41]
Traveller TCS	Superhuman in collaboration with human[42]	In both 1981 and 1982, Douglas Lenat's program Eurisko wins the US championship in Traveller TCS (a futuristic naval war game), prompting rule changes to block its unorthodox strategies.[43] Eurisko had heuristics for designing its fleet, and it also had heuristics for modifying its heuristics.
Othello	Superhuman	1997: The program Logistello wins every game in a six-game match against world champion Takeshi Murakami.[44]
Chess	Superhuman	1997: Deep Blue beats the world chess champion, Garry Kasparov. Kasparov claims to have seen glimpses of true intelligence and creativity in some of the computer's moves.[45] Since then, chess engines have continued to improve.[46]
Crosswords	Expert level	1999: The crossword-solving program Proverb outperforms the average crossword-solver.[47]

Table 1 *Continued*

		2012: The program Dr. Fill, created by Matt Ginsberg, scores in the top quartile among the otherwise human contestants in the American Crossword Puzzle Tournament. (Dr. Fill's performance is uneven. It completes perfectly the puzzle rated most difficult by humans, yet is stumped by a couple of nonstandard puzzles that involved spelling backwards or writing answers diagonally.)[48]
Scrabble	Superhuman	As of 2002, Scrabble-playing software surpasses the best human players.[49]
Bridge	Equal to the best	By 2005, contract bridge playing software reaches parity with the best human bridge players.[50]
Jeopardy!	Superhuman	2010: IBM's *Watson* defeats the two all-time-greatest human *Jeopardy!* champions, Ken Jennings and Brad Rutter.[51] *Jeopardy!* is a televised game show with trivia questions about history, literature, sports, geography, pop culture, science, and other topics. Questions are presented in the form of clues, and often involve wordplay.
Poker	Varied	Computer poker players remain slightly below the best humans for full-ring Texas hold 'em but perform at a superhuman level in some poker variants.[52]
FreeCell	Superhuman	Heuristics evolved using genetic algorithms produce a solver for the solitaire game FreeCell (which in its generalized form is NP-complete) that is able to beat high-ranking human players.[53]
Go	Very strong amateur level	As of 2012, the Zen series of go-playing programs has reached rank 6 dan in fast games (the level of a very strong amateur player), using Monte Carlo tree search and machine learning techniques.[54] Go-playing programs have been improving at a rate of about 1 dan/year in recent years. If this rate of improvement continues, they might beat the human world champion in about a decade.

machine, one would seem to have penetrated to the core of human intellectual endeavor."[55] This no longer seems so. One sympathizes with John McCarthy, who lamented: "As soon as it works, no one calls it AI anymore."[56]

There is an important sense, however, in which chess-playing AI turned out to be a lesser triumph than many imagined it would be. It was once supposed,

perhaps not unreasonably, that in order for a computer to play chess at grandmaster level, it would have to be endowed with a high degree of *general* intelligence.[57] One might have thought, for example, that great chess playing requires being able to learn abstract concepts, think cleverly about strategy, compose flexible plans, make a wide range of ingenious logical deductions, and maybe even model one's opponent's thinking. Not so. It turned out to be possible to build a perfectly fine chess engine around a special-purpose algorithm.[58] When implemented on the fast processors that became available towards the end of the twentieth century, it produces very strong play. But an AI built like that is narrow. It plays chess; it can do no other.[59]

In other domains, solutions have turned out to be *more* complicated than initially expected, and progress slower. The computer scientist Donald Knuth was struck that "AI has by now succeeded in doing essentially everything that requires 'thinking' but has failed to do most of what people and animals do 'without thinking'—that, somehow, is much harder!"[60] Analyzing visual scenes, recognizing objects, or controlling a robot's behavior as it interacts with a natural environment has proved challenging. Nevertheless, a fair amount of progress has been made and continues to be made, aided by steady improvements in hardware.

Common sense and natural language understanding have also turned out to be difficult. It is now often thought that achieving a fully human-level performance on these tasks is an "AI-complete" problem, meaning that the difficulty of solving these problems is essentially equivalent to the difficulty of building generally human-level intelligent machines.[61] In other words, if somebody *were* to succeed in creating an AI that could understand natural language as well as a human adult, they would in all likelihood also either already have succeeded in creating an AI that could do everything else that human intelligence can do, or they would be but a very short step from such a general capability.[62]

Chess-playing expertise turned out to be achievable by means of a surprisingly simple algorithm. It is tempting to speculate that other capabilities—such as general reasoning ability, or some key ability involved in programming—might likewise be achievable through some surprisingly simple algorithm. The fact that the best performance at one time is attained through a complicated mechanism does not mean that no simple mechanism could do the job as well or better. It might simply be that nobody has yet found the simpler alternative. The Ptolemaic system (with the Earth in the center, orbited by the Sun, the Moon, planets, and stars) represented the state of the art in astronomy for over a thousand years, and its predictive accuracy was improved over the centuries by progressively complicating the model: adding epicycles upon epicycles to the postulated celestial motions. Then the entire system was overthrown by the heliocentric theory of Copernicus, which was simpler and—though only after further elaboration by Kepler—more predictively accurate.[63]

Artificial intelligence methods are now used in more areas than it would make sense to review here, but mentioning a sampling of them will give an idea of the breadth of applications. Aside from the game AIs listed in Table 1, there

are hearing aids with algorithms that filter out ambient noise; route-finders that display maps and offer navigation advice to drivers; recommender systems that suggest books and music albums based on a user's previous purchases and ratings; and medical decision support systems that help doctors diagnose breast cancer, recommend treatment plans, and aid in the interpretation of electrocardiograms. There are robotic pets and cleaning robots, lawn-mowing robots, rescue robots, surgical robots, and over a million industrial robots.[64] The world population of robots exceeds 10 million.[65]

Modern speech recognition, based on statistical techniques such as hidden Markov models, has become sufficiently accurate for practical use (some fragments of this book were drafted with the help of a speech recognition program). Personal digital assistants, such as Apple's Siri, respond to spoken commands and can answer simple questions and execute commands. Optical character recognition of handwritten and typewritten text is routinely used in applications such as mail sorting and digitization of old documents.[66]

Machine translation remains imperfect but is good enough for many applications. Early systems used the GOFAI approach of hand-coded grammars that had to be developed by skilled linguists from the ground up for each language. Newer systems use statistical machine learning techniques that automatically build statistical models from observed usage patterns. The machine infers the parameters for these models by analyzing bilingual corpora. This approach dispenses with linguists: the programmers building these systems need not even speak the languages they are working with.[67]

Face recognition has improved sufficiently in recent years that it is now used at automated border crossings in Europe and Australia. The US Department of State operates a face recognition system with over 75 million photographs for visa processing. Surveillance systems employ increasingly sophisticated AI and data-mining technologies to analyze voice, video, or text, large quantities of which are trawled from the world's electronic communications media and stored in giant data centers.

Theorem-proving and equation-solving are by now so well established that they are hardly regarded as AI anymore. Equation solvers are included in scientific computing programs such as Mathematica. Formal verification methods, including automated theorem provers, are routinely used by chip manufacturers to verify the behavior of circuit designs prior to production.

The US military and intelligence establishments have been leading the way to the large-scale deployment of bomb-disposing robots, surveillance and attack drones, and other unmanned vehicles. These still depend mainly on remote control by human operators, but work is underway to extend their autonomous capabilities.

Intelligent scheduling is a major area of success. The DART tool for automated logistics planning and scheduling was used in Operation Desert Storm in 1991 to such effect that DARPA (the Defense Advanced Research Projects Agency in the United States) claims that this single application more than paid back their

thirty-year investment in AI.[68] Airline reservation systems use sophisticated scheduling and pricing systems. Businesses make wide use of AI techniques in inventory control systems. They also use automatic telephone reservation systems and helplines connected to speech recognition software to usher their hapless customers through labyrinths of interlocking menu options.

AI technologies underlie many Internet services. Software polices the world's email traffic, and despite continual adaptation by spammers to circumvent the countermeasures being brought against them, Bayesian spam filters have largely managed to hold the spam tide at bay. Software using AI components is responsible for automatically approving or declining credit card transactions, and continuously monitors account activity for signs of fraudulent use. Information retrieval systems also make extensive use of machine learning. The Google search engine is, arguably, the greatest AI system that has yet been built.

Now, it must be stressed that the demarcation between artificial intelligence and software in general is not sharp. Some of the applications listed above might be viewed more as generic software applications rather than AI in particular—though this brings us back to McCarthy's dictum that when something works it is no longer called AI. A more relevant distinction for our purposes is that between systems that have a narrow range of cognitive capability (whether they be called "AI" or not) and systems that have more generally applicable problem-solving capacities. Essentially all the systems currently in use are of the former type: narrow. However, many of them contain components that might also play a role in future artificial general intelligence or be of service in its development—components such as classifiers, search algorithms, planners, solvers, and representational frameworks.

One high-stakes and extremely competitive environment in which AI systems operate today is the global financial market. Automated stock-trading systems are widely used by major investing houses. While some of these are simply ways of automating the execution of particular buy or sell orders issued by a human fund manager, others pursue complicated trading strategies that adapt to changing market conditions. Analytic systems use an assortment of data-mining techniques and time series analysis to scan for patterns and trends in securities markets or to correlate historical price movements with external variables such as keywords in news tickers. Financial news providers sell newsfeeds that are specially formatted for use by such AI programs. Other systems specialize in finding arbitrage opportunities within or between markets, or in high-frequency trading that seeks to profit from minute price movements that occur over the course of milliseconds (a timescale at which communication latencies even for speed-of-light signals in optical fiber cable become significant, making it advantageous to locate computers near the exchange). Algorithmic high-frequency traders account for more than half of equity shares traded on US markets.[69] Algorithmic trading has been implicated in the 2010 Flash Crash (see Box 2).

Box 2 The 2010 Flash Crash

By the afternoon of May, 6, 2010, US equity markets were already down 4% on worries about the European debt crisis. At 2:32 p.m., a large seller (a mutual fund complex) initiated a sell algorithm to dispose of a large number of the E-Mini S&P 500 futures contracts to be sold off at a sell rate linked to a measure of minute-to-minute liquidity on the exchange. These contracts were bought by algorithmic high-frequency traders, which were programmed to quickly eliminate their temporary long positions by selling the contracts on to other traders. With demand from fundamental buyers slacking, the algorithmic traders started to sell the E-Minis primarily to other algorithmic traders, which in turn passed them on to other algorithmic traders, creating a "hot potato" effect driving up trading volume—this being interpreted by the sell algorithm as an indicator of high liquidity, prompting it to increase the rate at which it was putting E-Mini contracts on the market, pushing the downward spiral. At some point, the high-frequency traders started withdrawing from the market, drying up liquidity while prices continued to fall. At 2:45 p.m., trading on the E-Mini was halted by an automatic circuit breaker, the exchange's stop logic functionality. When trading was restarted, a mere five seconds later, prices stabilized and soon began to recover most of the losses. But for a while, at the trough of the crisis, a trillion dollars had been wiped off the market, and spillover effects had led to a substantial number of trades in individual securities being executed at "absurd" prices, such as one cent or 100,000 dollars. After the market closed for the day, representatives of the exchanges met with regulators and decided to break all trades that had been executed at prices 60% or more away from their pre-crisis levels (deeming such transactions "clearly erroneous" and thus subject to *post facto* cancellation under existing trade rules).[70]

The retelling here of this episode is a digression because the computer programs involved in the Flash Crash were not particularly intelligent or sophisticated, and the kind of threat they created is fundamentally different from the concerns we shall raise later in this book in relation to the prospect of machine superintelligence. Nevertheless, these events illustrate several useful lessons. One is the reminder that interactions between individually simple components (such as the sell algorithm and the high-frequency algorithmic trading programs) can produce complicated and unexpected effects. Systemic risk can build up in a system as new elements are introduced, risks that are not obvious until after something goes wrong (and sometimes not even then).[71]

Another lesson is that smart professionals might give an instruction to a program based on a sensible-seeming and normally sound assumption (e.g. that trading volume is a good measure of market liquidity), and that this can produce catastrophic results when the program continues to act on the instruction with iron-clad logical consistency even in the unanticipated situation where the assumption turns out to be invalid. The algorithm just does what it does; and unless

continued

Box 2 *Continued*

it is a very special kind of algorithm, it does not care that we clasp our heads and gasp in dumbstruck horror at the absurd inappropriateness of its actions. This is a theme that we will encounter again.

A third observation in relation to the Flash Crash is that while automation contributed to the incident, it also contributed to its resolution. The pre-preprogrammed stop order logic, which suspended trading when prices moved too far out of whack, was set to execute automatically because it had been correctly anticipated that the triggering events could happen on a timescale too swift for humans to respond. The need for pre-installed and automatically executing safety functionality—as opposed to reliance on runtime human supervision—again foreshadows a theme that will be important in our discussion of machine superintelligence.[72]

Opinions about the future of machine intelligence

Progress on two major fronts—towards a more solid statistical and information-theoretic foundation for machine learning on the one hand, and towards the practical and commercial success of various problem-specific or domain-specific applications on the other—has restored to AI research some of its lost prestige. There may, however, be a residual cultural effect on the AI community of its earlier history that makes many mainstream researchers reluctant to align themselves with over-grand ambition. Thus Nils Nilsson, one of the old-timers in the field, complains that his present-day colleagues lack the boldness of spirit that propelled the pioneers of his own generation:

> Concern for "respectability" has had, I think, a stultifying effect on some AI researchers. I hear them saying things like, "AI used to be criticized for its flossiness. Now that we have made solid progress, let us not risk losing our respectability." One result of this conservatism has been increased concentration on "weak AI"—the variety devoted to providing aids to human thought—and away from "strong AI"—the variety that attempts to mechanize human-level intelligence.[73]

Nilsson's sentiment has been echoed by several others of the founders, including Marvin Minsky, John McCarthy, and Patrick Winston.[74]

The last few years have seen a resurgence of interest in AI, which might yet spill over into renewed efforts towards artificial *general* intelligence (what Nilsson calls "strong AI"). In addition to faster hardware, a contemporary project would benefit from the great strides that have been made in the many subfields of AI, in software engineering more generally, and in neighboring fields such as computational neuroscience. One indication of pent-up demand for quality information and education is shown in the response to the free online

offering of an introductory course in artificial intelligence at Stanford University in the fall of 2011, organized by Sebastian Thrun and Peter Norvig. Some 160,000 students from around the world signed up to take it (and 23,000 completed it).[75]

Expert opinions about the future of AI vary wildly. There is disagreement about timescales as well as about what forms AI might eventually take. Predictions about the future development of artificial intelligence, one recent study noted, "are as confident as they are diverse."[76]

Although the contemporary distribution of belief has not been very carefully measured, we can get a rough impression from various smaller surveys and informal observations. In particular, a series of recent surveys have polled members of several relevant expert communities on the question of when they expect "human-level machine intelligence" (HLMI) to be developed, defined as "one that can carry out most human professions at least as well as a typical human."[77] Results are shown in Table 2. The combined sample gave the following (median) estimate: 10% probability of HLMI by 2022, 50% probability by 2040, and 90% probability by 2075. (Respondents were asked to premiss their estimates on the assumption that "human scientific activity continues without major negative disruption.")

These numbers should be taken with some grains of salt: sample sizes are quite small and not necessarily representative of the general expert population. They are, however, in concordance with results from other surveys.[78]

The survey results are also in line with some recently published interviews with about two-dozen researchers in AI-related fields. For example, Nils Nilsson has spent a long and productive career working on problems in search, planning, knowledge representation, and robotics; he has authored textbooks in artificial intelligence; and he recently completed the most comprehensive history of the field written to date.[79] When asked about arrival dates for HLMI, he offered the following opinion:[80]

10% chance: 2030
50% chance: 2050
90% chance: 2100

Table 2 ***When will human-level machine intelligence be attained?***[81]

	10%	50%	90%
PT-AI	2023	2048	2080
AGI	2022	2040	2065
EETN	2020	2050	2093
TOP100	2024	2050	2070
Combined	2022	2040	2075

Judging from the published interview transcripts, Professor Nilsson's probability distribution appears to be quite representative of many experts in the area—though again it must be emphasized that there is a wide spread of opinion: there are practitioners who are substantially more boosterish, confidently expecting HLMI in the 2020–40 range, and others who are confident either that it will never happen or that it is indefinitely far off.[82] In addition, some interviewees feel that the notion of a "human level" of artificial intelligence is ill-defined or misleading, or are for other reasons reluctant to go on record with a quantitative prediction.

My own view is that the median numbers reported in the expert survey do not have enough probability mass on later arrival dates. A 10% probability of HLMI not having been developed by 2075 or even 2100 (after conditionalizing on "human scientific activity continuing without major negative disruption") seems too low.

Historically, AI researchers have not had a strong record of being able to predict the rate of advances in their own field or the shape that such advances would take. On the one hand, some tasks, like chess playing, turned out to be achievable by means of surprisingly simple programs; and naysayers who claimed that machines would "never" be able to do this or that have repeatedly been proven wrong. On the other hand, the more typical errors among practitioners have been to underestimate the difficulties of getting a system to perform robustly on real-world tasks, and to overestimate the advantages of their own particular pet project or technique.

The survey also asked two other questions of relevance to our inquiry. One inquired of respondents about how much longer they thought it would take to reach superintelligence assuming human-level machine is first achieved. The results are in Table 3.

Another question inquired what they thought would be the overall long-term impact for humanity of achieving human-level machine intelligence. The answers are summarized in Figure 2.

My own views again differ somewhat from the opinions expressed in the survey. I assign a higher probability to superintelligence being created relatively soon after human-level machine intelligence. I also have a more polarized outlook on the consequences, thinking an extremely good or an extremely bad outcome to be somewhat more likely than a more balanced outcome. The reasons for this will become clear later in the book.

Table 3 ***How long from human level to superintelligence?***

	Within 2 years after HLMI	Within 30 years after HLMI
TOP100	5%	50%
Combined	10%	75%

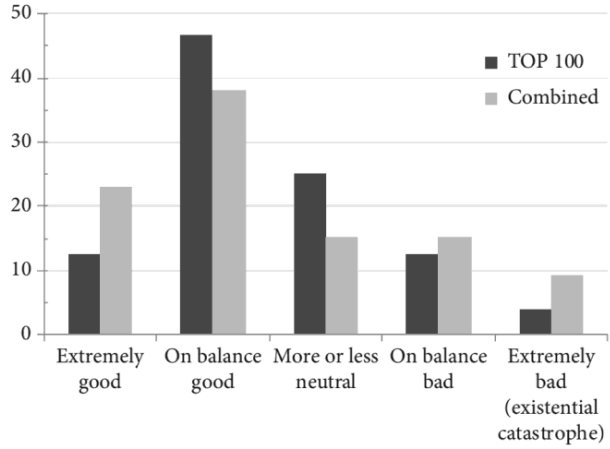


Figure 2 Overall long-term impact of HLMI.[83]

Small sample sizes, selection biases, and—above all—the inherent unreliability of the subjective opinions elicited mean that one should not read too much into these expert surveys and interviews. They do not let us draw any strong conclusion. But they do hint at a weak conclusion. They suggest that (at least in lieu of better data or analysis) it may be reasonable to believe that human-level machine intelligence has a fairly sizeable chance of being developed by mid-century, and that it has a non-trivial chance of being developed considerably sooner or much later; that it might perhaps fairly soon thereafter result in superintelligence; and that a wide range of outcomes may have a significant chance of occurring, including extremely good outcomes and outcomes that are as bad as human extinction.[84] At the very least, they suggest that the topic is worth a closer look.

CHAPTER 2

Paths to superintelligence

Machines are currently far inferior to humans in general intelligence. Yet one day (we have suggested) they will be superintelligent. How do we get from here to there? This chapter explores several conceivable technological paths. We look at artificial intelligence, whole brain emulation, biological cognition, and human–machine interfaces, as well as networks and organizations. We evaluate their different degrees of plausibility as pathways to superintelligence. The existence of multiple paths increases the probability that the destination can be reached via at least one of them.

We can tentatively define a superintelligence as *any intellect that greatly exceeds the cognitive performance of humans in virtually all domains of interest*.[1] We will have more to say about the concept of superintelligence in the next chapter, where we will subject it to a kind of spectral analysis to distinguish some different possible forms of superintelligence. But for now, the rough characterization just given will suffice. Note that the definition is noncommittal about how the superintelligence is implemented. It is also noncommittal regarding qualia: whether a superintelligence would have subjective conscious experience might matter greatly for some questions (in particular for some moral questions), but our primary focus here is on the causal antecedents and consequences of superintelligence, not on the metaphysics of mind.[2]

The chess program Deep Fritz is not a superintelligence on this definition, since Fritz is only smart within the narrow domain of chess. Certain kinds of domain-specific superintelligence could, however, be important. When referring to superintelligent performance limited to a particular domain, we will note the restriction explicitly. For instance, an "engineering superintelligence" would be an intellect that vastly outperforms the best current human minds in the domain of engineering. Unless otherwise noted, we use the term to refer to systems that have a superhuman level of *general* intelligence.

But how might we create superintelligence? Let us examine some possible paths.

Artificial intelligence

Readers of this chapter must not expect a blueprint for programming an artificial general intelligence. No such blueprint exists yet, of course. And had I been in possession of such a blueprint, I most certainly would not have published it in a book. (If the reasons for this are not immediately obvious, the arguments in subsequent chapters will make them clear.)

We can, however, discern some general features of the kind of system that would be required. It now seems clear that a capacity to learn would be an integral feature of the core design of a system intended to attain general intelligence, not something to be tacked on later as an extension or an afterthought. The same holds for the ability to deal effectively with uncertainty and probabilistic information. Some faculty for extracting useful concepts from sensory data and internal states, and for leveraging acquired concepts into flexible combinatorial representations for use in logical and intuitive reasoning, also likely belong among the core design features in a modern AI intended to attain general intelligence.

The early Good Old-Fashioned Artificial Intelligence systems did not, for the most part, focus on learning, uncertainty, or concept formation, perhaps because techniques for dealing with these dimensions were poorly developed at the time. This is not to say that the underlying ideas are all that novel. The idea of using learning as a means of bootstrapping a simpler system to human-level intelligence can be traced back at least to Alan Turing's notion of a "child machine," which he wrote about in 1950:

> Instead of trying to produce a programme to simulate the adult mind, why not rather try to produce one which simulates the child's? If this were then subjected to an appropriate course of education one would obtain the adult brain.[3]

Turing envisaged an iterative process to develop such a child machine:

> We cannot expect to find a good child machine at the first attempt. One must experiment with teaching one such machine and see how well it learns. One can then try another and see if it is better or worse. There is an obvious connection between this process and evolution. . . . One may hope, however, that this process will be more expeditious than evolution. The survival of the fittest is a slow method for measuring advantages. The experimenter, by the exercise of intelligence, should be able to speed it up. Equally important is the fact that he is not restricted to random mutations. If he can trace a cause for some weakness he can probably think of the kind of mutation which will improve it.[4]

We know that blind evolutionary processes can produce human-level general intelligence, since they have already done so at least once. Evolutionary processes with foresight—that is, genetic programs designed and guided by an intelligent human programmer—should be able to achieve a similar outcome with far greater efficiency. This observation has been used by some philosophers and scientists,

including David Chalmers and Hans Moravec, to argue that human-level AI is not only theoretically possible but feasible within this century.[5] The idea is that we can estimate the relative capabilities of evolution and human engineering to produce intelligence, and find that human engineering is already vastly superior to evolution in some areas and is likely to become superior in the remaining areas before too long. The fact that evolution produced intelligence therefore indicates that human engineering will soon be able to do the same. Thus, Moravec wrote (already back in 1976):

> The existence of several examples of intelligence designed under these constraints should give us great confidence that we can achieve the same in short order. The situation is analogous to the history of heavier than air flight, where birds, bats and insects clearly demonstrated the possibility before our culture mastered it.[6]

One needs to be cautious, though, in what inferences one draws from this line of reasoning. It is true that evolution produced heavier-than-air flight, and that human engineers subsequently succeeded in doing likewise (albeit by means of a very different mechanism). Other examples could also be adduced, such as sonar, magnetic navigation, chemical weapons, photoreceptors, and all kinds of mechanic and kinetic performance characteristics. However, one could equally point to areas where human engineers have thus far failed to match evolution: in morphogenesis, self-repair, and the immune defense, for example, human efforts lag far behind what nature has accomplished. Moravec's argument, therefore, cannot give us "great confidence" that we can achieve human-level artificial intelligence "in short order." At best, the evolution of intelligent life places an upper bound on the intrinsic difficulty of designing intelligence. But this upper bound could be quite far above current human engineering capabilities.

Another way of deploying an evolutionary argument for the feasibility of AI is via the idea that we could, by running genetic algorithms on sufficiently fast computers, achieve results comparable to those of biological evolution. This version of the evolutionary argument thus proposes a specific method whereby intelligence could be produced.

But is it true that we will soon have computing power sufficient to recapitulate the relevant evolutionary processes that produced human intelligence? The answer depends both on how much computing technology will advance over the next decades and on how much computing power would be required to run genetic algorithms with the same optimization power as the evolutionary process of natural selection that lies in our past. Although, in the end, the conclusion we get from pursuing this line of reasoning is disappointingly indeterminate, it is instructive to attempt a rough estimate (see Box 3). If nothing else, the exercise draws attention to some interesting unknowns.

The upshot is that the computational resources required to simply replicate the relevant evolutionary processes on Earth that produced human-level intelligence are severely out of reach—and will remain so even if Moore's law were to continue

Box 3 What would it take to recapitulate evolution?

Not every feat accomplished by evolution in the course of the development of human intelligence is relevant to a human engineer trying to artificially evolve machine intelligence. Only a small portion of evolutionary selection on Earth has been selection for intelligence. More specifically, the problems that human engineers cannot trivially bypass may have been the target of a very small portion of total evolutionary selection. For example, since we can run our computers on electrical power, we do not have to reinvent the molecules of the cellular energy economy in order to create intelligent machines—yet such molecular evolution of metabolic pathways might have used up a large part of the total amount of selection power that was available to evolution over the course of Earth's history.[7]

One might argue that the key insights for AI are embodied in the structure of nervous systems, which came into existence less than a billion years ago.[8] If we take that view, then the number of relevant "experiments" available to evolution is drastically curtailed. There are some $4–6\times10^{30}$ prokaryotes in the world today, but only 10^{19} insects, and fewer than 10^{10} humans (while pre-agricultural populations were orders of magnitude smaller).[9] These numbers are only moderately intimidating.

Evolutionary algorithms, however, require not only variations to select among but also a fitness function to evaluate variants, and this is typically the most computationally expensive component. A fitness function for the evolution of artificial intelligence plausibly requires simulation of neural development, learning, and cognition to evaluate fitness. We might thus do better not to look at the raw number of organisms with complex nervous systems, but instead to attend to the number of neurons in biological organisms that we might need to simulate to mimic evolution's fitness function. We can make a crude estimate of that latter quantity by considering insects, which dominate terrestrial animal biomass (with ants alone estimated to contribute some 15–20%).[10] Insect brain size varies substantially, with large and social insects sporting larger brains: a honeybee brain has just under 10^6 neurons, a fruit fly brain has 10^5 neurons, and ants are in between with 250,000 neurons.[11] The majority of smaller insects may have brains of only a few thousand neurons. Erring on the side of conservatively high, if we assigned all 10^{19} insects fruit-fly numbers of neurons, the total would be 10^{24} insect neurons in the world. This could be augmented with an additional order of magnitude to account for aquatic copepods, birds, reptiles, mammals, etc., to reach 10^{25}. (By contrast, in pre-agricultural times there were fewer than 10^7 humans, with under 10^{11} neurons each: thus fewer than 10^{18} human neurons in total, though humans have a higher number of synapses per neuron.)

The computational cost of simulating one neuron depends on the level of detail that one includes in the simulation. Extremely simple neuron models use about 1,000 floating-point operations per second (FLOPS) to simulate one neuron (in real-time). The electrophysiologically realistic Hodgkin–Huxley model

continued

Box 3 *Continued*

uses 1,200,000 FLOPS. A more detailed multi-compartmental model would add another three to four orders of magnitude, while higher-level models that abstract systems of neurons could subtract two to three orders of magnitude from the simple models.[12] If we were to simulate 10^{25} neurons over a billion years of evolution (longer than the existence of nervous systems as we know them), and we allow our computers to run for one year, these figures would give us a requirement in the range of 10^{31}–10^{44} FLOPS. For comparison, China's Tianhe-2, the world's most powerful supercomputer as of September 2013, provides only 3.39×10^{16} FLOPS. In recent decades, it has taken approximately 6.7 years for commodity computers to increase in power by one order of magnitude. Even a century of continued Moore's law would not be enough to close this gap. Running more specialized hardware, or allowing longer run-times, could contribute only a few more orders of magnitude.

This figure is conservative in another respect. Evolution achieved human intelligence without aiming at this outcome. In other words, the fitness functions for natural organisms do not select only for intelligence and its precursors.[13] Even environments in which organisms with superior information processing skills reap various rewards may not select for intelligence, because improvements to intelligence can (and often do) impose significant costs, such as higher energy consumption or slower maturation times, and those costs may outweigh whatever benefits are gained from smarter behavior. Excessively deadly environments also reduce the value of intelligence: the shorter one's expected lifespan, the less time there will be for increased learning ability to pay off. Reduced selective pressure for intelligence slows the spread of intelligence-enhancing innovations, and thus the opportunity for selection to favor subsequent innovations that depend on them. Furthermore, evolution may wind up stuck in local optima that humans would notice and bypass by altering tradeoffs between exploitation and exploration or by providing a smooth progression of increasingly difficult intelligence tests.[14] And as mentioned earlier, evolution scatters much of its selection power on traits that are unrelated to intelligence (such as Red Queen's races of competitive co-evolution between immune systems and parasites). Evolution continues to waste resources producing mutations that have proved consistently lethal, and it fails to take advantage of statistical similarities in the effects of different mutations. These are all inefficiencies in natural selection (when viewed as a means of evolving intelligence) that it would be relatively easy for a human engineer to avoid while using evolutionary algorithms to develop intelligent software.

It is plausible that eliminating inefficiencies like those just described would trim many orders of magnitude off the 10^{31}–10^{44} FLOPS range calculated earlier. Unfortunately, it is difficult to know how many orders of magnitude. It is difficult even to make a rough estimate—for aught we know, the efficiency savings could be five orders of magnitude, or ten, or twenty-five.[15]

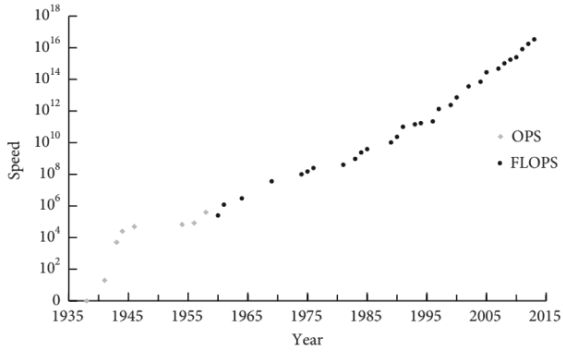


Figure 3 Supercomputer performance. In a narrow sense, "Moore's law" refers to the observation that the number of transistors on integrated circuits have for several decades doubled approximately every two years. However, the term is often used to refer to the more general observation that many performance metrics in computing technology have followed a similarly fast exponential trend. Here we plot peak speed of the world's fastest supercomputer as a function of time (on a logarithmic vertical scale). In recent years, growth in the serial speed of processors has stagnated, but increased use of parallelization has enabled the total number of computations performed to remain on the trend line.[16]

for a century (cf. Figure 3). It is plausible, however, that compared with brute-force replication of natural evolutionary processes, vast efficiency gains are achievable by designing the search process to *aim* for intelligence, using various obvious improvements over natural selection. Yet it is very hard to bound the magnitude of those attainable efficiency gains. We cannot even say whether they amount to five or to twenty-five orders of magnitude. Absent further elaboration, therefore, evolutionary arguments are not able to meaningfully constrain our expectations of either the difficulty of building human-level machine intelligence or the timescales for such developments.

There is a further complication with these kinds of evolutionary considerations, one that makes it hard to derive from them even a very loose upper bound on the difficulty of evolving intelligence. We must avoid the error of inferring, from the fact that intelligent life evolved on Earth, that the evolutionary processes involved had a reasonably high prior probability of producing intelligence. Such an inference is unsound because it fails to take account of the observation selection effect that guarantees that all observers will find themselves having originated on a planet where intelligent life arose, no matter how likely or unlikely it was for any given such planet to produce intelligence. Suppose, for example, that in addition to the systematic effects of natural selection it required an enormous amount of *lucky coincidence* to produce intelligent life—enough so that intelligent life evolves on only one planet out of every 10^{30} planets on which simple replicators arise. In that case, when we run our genetic algorithms to try to replicate what natural evolution did, we might find that we must run some 10^{30} simulations before we find one where all the elements come together in just the right way. This seems fully consistent with our observation that life did evolve here on Earth. Only by

careful and somewhat intricate reasoning—by analyzing instances of convergent evolution of intelligence-related traits and engaging with the subtleties of observation selection theory—can we partially circumvent this epistemological barrier. Unless one takes the trouble to do so, one is not in a position to rule out the possibility that the alleged "upper bound" on the computational requirements for recapitulating the evolution of intelligence derived in Box 3 might be too low by thirty orders of magnitude (or some other such large number).[17]

Another way of arguing for the feasibility of artificial intelligence is by pointing to the human brain and suggesting that we could use it as a template for a machine intelligence. One can distinguish different versions of this approach based on how closely they propose to imitate biological brain functions. At one extreme—that of very close imitation—we have the idea of *whole brain emulation*, which we will discuss in the next subsection. At the other extreme are approaches that take their inspiration from the functioning of the brain but do not attempt low-level imitation. Advances in neuroscience and cognitive psychology—which will be aided by improvements in instrumentation—should eventually uncover the general principles of brain function. This knowledge could then guide AI efforts. We have already encountered neural networks as an example of a brain-inspired AI technique. Hierarchical perceptual organization is another idea that has been transferred from brain science to machine learning. The study of reinforcement learning has been motivated (at least in part) by its role in psychological theories of animal cognition, and reinforcement learning techniques (e.g. the "TD-algorithm") inspired by these theories are now widely used in AI.[18] More cases like these will surely accumulate in the future. Since there is a limited number—perhaps a very small number—of distinct fundamental mechanisms that operate in the brain, continuing incremental progress in brain science should eventually discover them all. Before this happens, though, it is possible that a hybrid approach, combining some brain-inspired techniques with some purely artificial methods, would cross the finishing line. In that case, the resultant system need not be recognizably brain-like even though some brain-derived insights were used in its development.

The availability of the brain as template provides strong support for the claim that machine intelligence is ultimately feasible. This, however, does not enable us to predict when it will be achieved because it is hard to predict the future rate of discoveries in brain science. What we can say is that the further into the future we look, the greater the likelihood that the secrets of the brain's functionality will have been decoded sufficiently to enable the creation of machine intelligence in this manner.

Different people working toward machine intelligence hold different views about how promising neuromorphic approaches are compared with approaches that aim for completely synthetic designs. The existence of birds demonstrated that heavier-than-air flight was physically possible and prompted efforts to build flying machines. Yet the first functioning airplanes did not flap their wings. The jury is out on whether machine intelligence will be like flight, which humans

achieved through an artificial mechanism, or like combustion, which we initially mastered by copying naturally occurring fires.

Turing's idea of designing a program that acquires most of its content by learning, rather than having it pre-programmed at the outset, can apply equally to neuromorphic and synthetic approaches to machine intelligence.

A variation on Turing's conception of a child machine is the idea of a "seed AI."[19] Whereas a child machine, as Turing seems to have envisaged it, would have a relatively fixed architecture that simply develops its inherent potentialities by accumulating *content*, a seed AI would be a more sophisticated artificial intelligence capable of improving its own *architecture*. In the early stages of a seed AI, such improvements might occur mainly through trial and error, information acquisition, or assistance from the programmers. At its later stages, however, a seed AI should be able to *understand* its own workings sufficiently to engineer new algorithms and computational structures to bootstrap its cognitive performance. This needed understanding could result from the seed AI reaching a sufficient level of general intelligence across many domains, or from crossing some threshold in a particularly relevant domain such as computer science or mathematics.

This brings us to another important concept, that of "recursive self-improvement." A successful seed AI would be able to iteratively enhance itself: an early version of the AI could design an improved version of itself, and the improved version—being smarter than the original—might be able to design an even smarter version of itself, and so forth.[20] Under some conditions, such a process of recursive self-improvement might continue long enough to result in an intelligence explosion—an event in which, in a short period of time, a system's level of intelligence increases from a relatively modest endowment of cognitive capabilities (perhaps sub-human in most respects, but with a domain-specific talent for coding and AI research) to radical superintelligence. We will return to this important possibility in Chapter 4, where the dynamics of such an event will be analyzed more closely. Note that this model suggests the possibility of surprises: attempts to build artificial general intelligence might fail pretty much completely until the last missing critical component is put in place, at which point a seed AI might become capable of sustained recursive self-improvement.

Before we end this subsection, there is one more thing that we should emphasize, which is that an artificial intelligence need not much resemble a human mind. AIs could be—indeed, it is likely that most will be—extremely alien. We should expect that they will have very different cognitive architectures than biological intelligences, and in their early stages of development they will have very different profiles of cognitive strengths and weaknesses (though, as we shall later argue, they could eventually overcome any initial weakness). Furthermore, the goal systems of AIs could diverge radically from those of human beings. There is no reason to expect a generic AI to be motivated by love or hate or pride or other such common human sentiments: these complex adaptations would require deliberate expensive effort to recreate in AIs. This is at once a big problem and a big opportunity.

We will return to the issue of AI motivation in later chapters, but it is so central to the argument in this book that it is worth bearing in mind throughout.

Whole brain emulation

In whole brain emulation (also known as "uploading"), intelligent software would be produced by scanning and closely modeling the computational structure of a biological brain. This approach thus represents a limiting case of drawing inspiration from nature: barefaced plagiarism. Achieving whole brain emulation requires the accomplishment of the following steps.

First, a sufficiently detailed scan of a particular human brain is created. This might involve stabilizing the brain post-mortem through vitrification (a process that turns tissue into a kind of glass). A machine could then dissect the tissue into thin slices, which could be fed into another machine for scanning, perhaps by an array of electron microscopes. Various stains might be applied at this stage to bring out different structural and chemical properties. Many scanning machines could work in parallel to process multiple brain slices simultaneously.

Second, the raw data from the scanners is fed to a computer for automated image processing to reconstruct the three-dimensional neuronal network that implemented cognition in the original brain. In practice, this step might proceed concurrently with the first step to reduce the amount of high-resolution image data stored in buffers. The resulting map is then combined with a library of neurocomputational models of different types of neurons or of different neuronal elements (such as particular kinds of synaptic connectors). Figure 4 shows some results of scanning and image processing produced with present-day technology.

In the third stage, the neurocomputational structure resulting from the previous step is implemented on a sufficiently powerful computer. If completely successful, the result would be a digital reproduction of the original intellect, with memory and personality intact. The emulated human mind now exists as software on a computer. The mind can either inhabit a virtual reality or interface with the external world by means of robotic appendages.

The whole brain emulation path does not require that we figure out how human cognition works or how to program an artificial intelligence. It requires only that we understand the low-level functional characteristics of the basic computational elements of the brain. No fundamental conceptual or theoretical breakthrough is needed for whole brain emulation to succeed.

Whole brain emulation does, however, require some rather advanced enabling technologies. There are three key prerequisites: (1) *scanning*: high-throughput microscopy with sufficient resolution and detection of relevant properties; (2) *translation*: automated image analysis to turn raw scanning data into an interpreted three-dimensional model of relevant neurocomputational elements; and (3) *simulation*: hardware powerful enough to implement the resultant computational structure (see Table 4). (In comparison with these more challenging

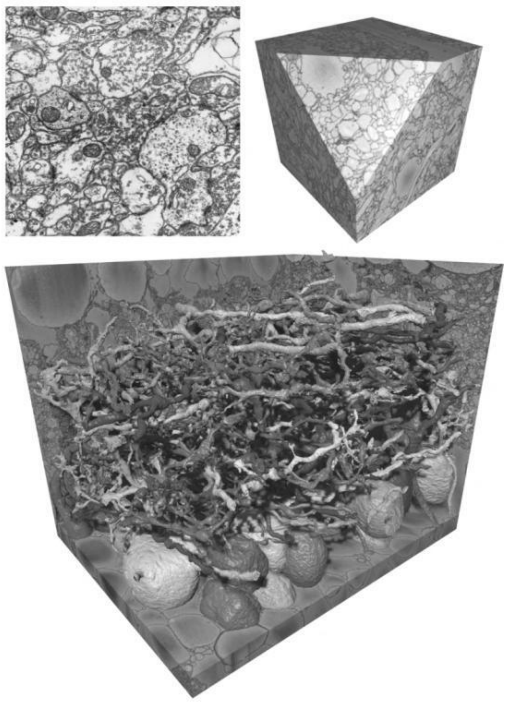


Figure 4 Reconstructing 3D neuroanatomy from electron microscope images. *Upper left*: A typical electron micrograph showing cross-sections of neuronal matter—dendrites and axons. *Upper right*: Volume image of rabbit retinal neural tissue acquired by serial block-face scanning electron microscopy.[21] Individual 2D images have been stacked into a cube (with a side of approximately 11 μm). *Bottom*: Reconstruction of a subset of the neuronal projections filling a volume of neuropil, generated by an automated segmentation algorithm.[22]

steps, the construction of a basic virtual reality or a robotic embodiment with an audiovisual input channel and some simple output channel is relatively easy. Simple yet minimally adequate I/O seems feasible already with present technology.[23])

There is good reason to think that the requisite enabling technologies are attainable, though not in the near future. Reasonable computational models of many types of neuron and neuronal processes already exist. Image recognition software has been developed that can trace axons and dendrites through a stack of two-dimensional images (though reliability needs to be improved). And there are imaging tools that provide the necessary resolution—with a scanning tunneling microscope it is possible to "see" individual atoms, which is a far higher resolution than needed. However, although present knowledge and capabilities suggest that there is no in-principle barrier to the development of the requisite enabling technologies, it is clear that a very great deal of incremental technical progress would be needed to bring human whole brain emulation within reach.[24] For example, microscopy technology would need not just sufficient resolution but also sufficient throughput. Using an atomic-resolution scanning tunneling microscope to image the needed surface area would be far too slow to be practicable. It would be more plausible to use a lower-resolution electron microscope, but this would require new methods for