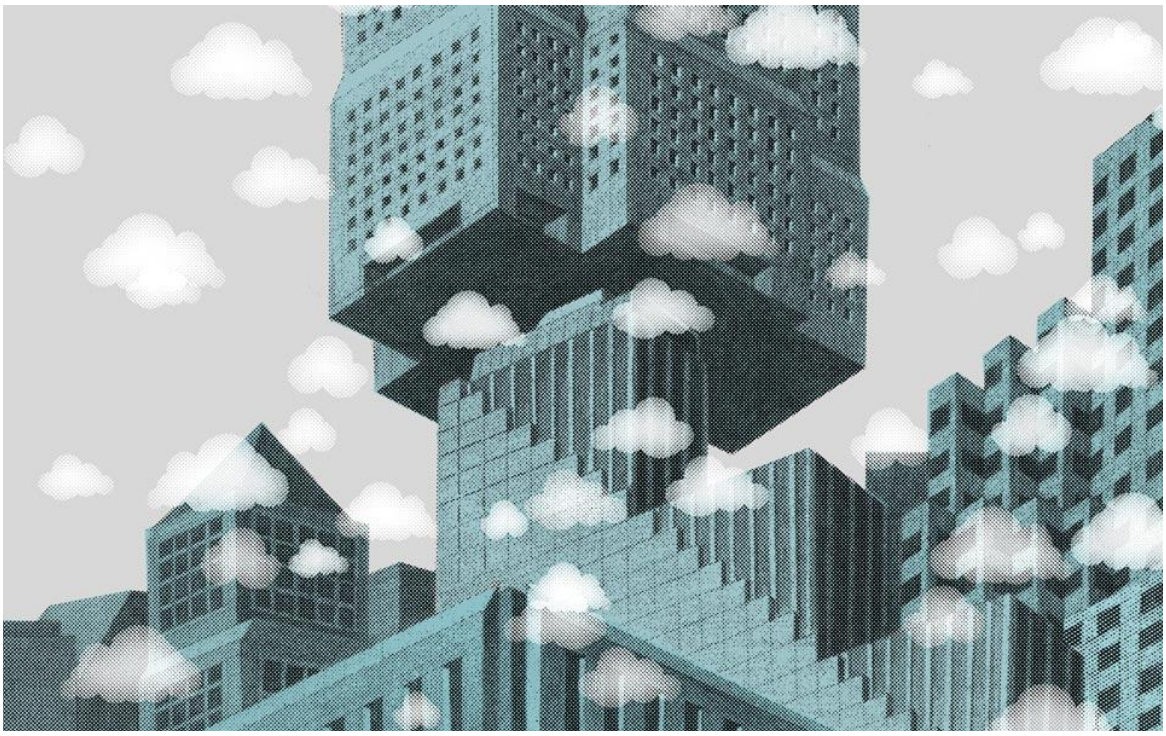
Creative blocks

The very laws of physics imply that artificial intelligence must be possible. What's holding us up?

David Deutsch 03 October 2012



'Expecting to create an AGI without first understanding how it works is like expecting skyscrapers to fly if we build them tall enough.' *Illustration by Sam Green*

David Deutsch is a physicist at the University of Oxford and a fellow of the Royal Society. His latest book is The Beginning of Infinity.

It is uncontroversial that the human brain has capabilities that are, in some respects, far superior to those of all other known objects in the cosmos. It is the only kind of object capable of understanding that the cosmos is even there, or why there are infinitely many prime numbers, or that apples fall because of the curvature of space-time, or that obeying its own inborn instincts can be morally wrong, or that it itself exists. Nor are its unique abilities confined to such cerebral matters. The cold, physical fact is that it is the only kind of object that can propel itself into space and back without harm, or predict and prevent a meteor strike on itself, or cool objects to a billionth of a degree above absolute zero, or detect others of its kind across galactic distances.

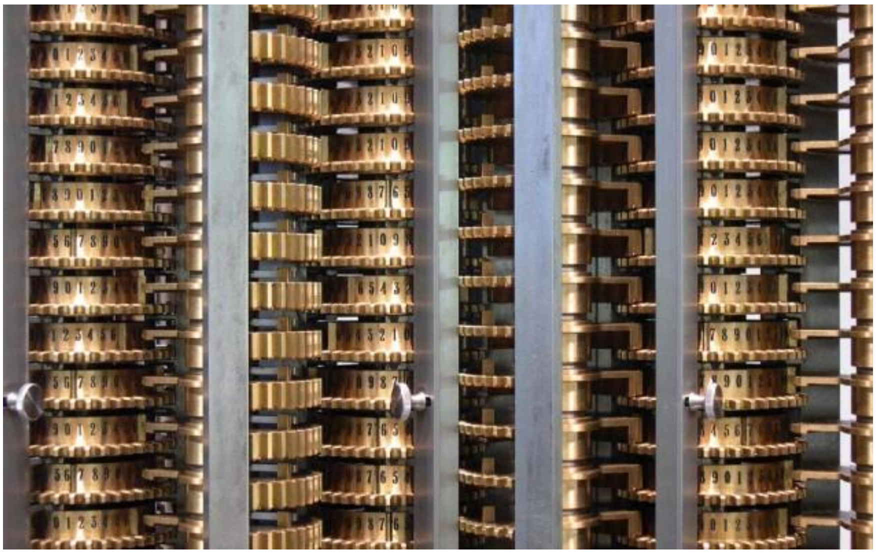
But no brain on Earth is yet close to knowing what brains *do* in order to achieve any of that functionality. The enterprise of achieving it artificially — the field of ‘artificial general intelligence’ or AGI — has made no progress whatever during the entire six decades of its existence.

Why? Because, as an unknown sage once remarked, ‘it ain’t what we don’t know that causes trouble, it’s what we know for sure that just ain’t so’ (and if you know that sage was Mark Twain, then what you know ain’t so either). I cannot think of any other significant field of knowledge in which the prevailing wisdom, not only in society at large but also among experts, is so beset with entrenched, overlapping, fundamental errors. Yet it has also been one of the most self-confident fields in prophesying that it will soon achieve the ultimate breakthrough.

Despite this long record of failure, *AGI must be possible*. And that is because of a deep property of the laws of physics, namely the*universality of computation*. This entails that everything that the laws of physics require a physical object to do can, in principle, be emulated in arbitrarily fine detail by some program on a general-purpose computer, provided it is given enough time and memory. The first people to guess this and to grapple with its ramifications were the 19th-century mathematician Charles Babbage and his assistant Ada, Countess of Lovelace. It remained a guess until the 1980s, when I proved it using the quantum theory of computation.

Babbage came upon universality from an unpromising direction. He had been much exercised by the fact that tables of mathematical functions (such as logarithms and cosines) contained mistakes. At the time they were compiled by armies of clerks, known as ‘computers’, which is the origin of the word. Being human, the computers were fallible. There were elaborate systems of error correction, but even proofreading for typographical errors was a nightmare. Such errors were not merely inconvenient and expensive: they could cost lives. For instance, the tables were extensively used in navigation. So, Babbage designed a mechanical calculator, which he called the*Difference Engine*. It would be programmed by initialising certain cogs. The mechanism would drive a printer, in order to automate the production of the tables. That would bring the error rate down to negligible levels, to the eternal benefit of humankind.

Unfortunately, Babbage’s project-management skills were so poor that despite spending vast amounts of his own and the British government’s money, he never managed to get the machine built. Yet his design was sound, and has since been implemented by a team led by the engineer Doron Swade at the Science Museum in London.



Slow but steady: a detail from Charles Babbage's Difference Engine, assembled nearly 170 years after it was designed. *Courtesy Science Museum*

Here was a cognitive task that only humans had been able to perform. Nothing else in the known universe even came close to matching them, but the Difference Engine would perform *better* than the best humans. And therefore, even at that faltering, embryonic stage of the history of automated computation — before Babbage had considered anything like AGI — we can see the seeds of a philosophical puzzle that is controversial to this day: what exactly is the difference between what the human 'computers' were doing and what the Difference Engine could do? What type of cognitive task, if any, could either type of entity perform that the other could not in principle perform too?

One immediate difference between them was that the sequence of elementary steps (of counting, adding, multiplying by 10, and so on) that the Difference Engine used to compute a given function did not mirror those of the human 'computers'. That is to say, they used different algorithms. In itself, that is not a fundamental difference: the Difference Engine could have been modified with additional gears and levers to mimic the humans' algorithm exactly. Yet that would have achieved nothing except an increase in the error rate, due to increased numbers of glitches in the more complex machinery. Similarly, the humans, given different instructions but no hardware changes, would have been capable of emulating every detail of the Difference Engine's method — and doing so would have been just as perverse. It would not have copied the Engine's main advantage, its accuracy, which was due to hardware not software. It would only have made an arduous, boring task even more arduous and boring, which would have made errors more likely, not less.

Babbage knew that it could be programmed to do algebra, play chess, compose music, process images and so on

For humans, that difference in outcomes — the different error rate — would have been caused by the fact that computing exactly the same table with two different algorithms *felt different*. But it would not have felt different to the Difference Engine. It had no feelings. Experiencing boredom was one of many cognitive tasks at which the Difference Engine would have been hopelessly inferior to humans. Nor was it capable of *knowing* or *proving*, as Babbage did, that the two algorithms would give identical results if executed accurately. Still less was it capable of *wanting*, as he did, to benefit seafarers and humankind in general. In fact, its repertoire was confined to evaluating a tiny class of specialised mathematical functions (basically, power series in a single variable).

Thinking about how he could enlarge that repertoire, Babbage first realised that the programming phase of the Engine's operation could itself be automated: the initial settings of the cogs could be encoded on punched cards. And then he had an epoch-making insight. The Engine could be adapted to punch new cards and store them for its own later use, making what we today call a computer memory. If it could run for long enough — powered, as he envisaged, by a steam engine — and had an unlimited supply of blank cards, its repertoire would jump from that tiny class of mathematical functions to the set of *all* computations that can possibly be performed by *any physical object*. That's universality.

Babbage called this improved machine the *Analytical Engine*. He and Lovelace understood that its universality would give it revolutionary potential to improve almost every scientific endeavour and manufacturing process, as well as everyday life. They showed remarkable foresight about specific applications. They knew that it could be programmed to do algebra, play chess, compose music, process images and so on. Unlike the Difference Engine, it *could* be programmed to use exactly the same method as humans used to make those tables. *And* prove that the two methods must give the same answers, *and* do the same error-checking and proofreading (using, say, optical character recognition) as well.

But could the Analytical Engine feel the same boredom? Could it feel anything? Could it want to better the lot of humankind (or of Analytical Enginekind)? Could it disagree with its programmer about its programming? Here is where Babbage and Lovelace's insight failed them. They thought that *some* cognitive functions of the human brain were beyond the reach of computational universality. As Lovelace wrote, 'The Analytical Engine has no pretensions whatever to originate anything. It can do whatever we know how to order it to perform. It can follow analysis; but it has no power of anticipating any analytical relations or truths.'

And yet 'originating things', 'following analysis', and 'anticipating analytical relations and truths' are all behaviours of brains and, therefore, of the atoms of which brains are composed. Such behaviours obey the laws of physics. So it follows inexorably from universality that, with the right program, an Analytical Engine would undergo them too, atom by atom and step by step. True, the atoms in the brain would be emulated by metal cogs and levers rather than organic material — but in the present context, inferring anything substantive from that distinction would be rank racism.

Despite their best efforts, Babbage and Lovelace failed almost entirely to convey their enthusiasm about the Analytical Engine to others. In one of the great might-have-beens of history, the idea of a universal computer languished on the back burner of human thought. There it remained until the 20th century, when Alan Turing arrived with a spectacular series of intellectual *tours de force*, laying the foundations of the classical theory of computation, establishing the limits of computability, participating in the building of the first universal classical computer and, by helping to crack the Enigma code, contributing to the Allied victory in the Second World War.

Turing fully understood universality. In his 1950 paper 'Computing Machinery and Intelligence', he used it to sweep away what he called 'Lady Lovelace's objection', and every other objection both reasonable and unreasonable. He concluded that a computer program whose repertoire included *all* the distinctive attributes of the human brain — feelings, free will, consciousness and all — could be written.

This astounding claim split the intellectual world into two camps, one insisting that AGI was none the less impossible, and the other that it was imminent. Both were mistaken. The first, initially predominant, camp cited a plethora of reasons ranging from the supernatural to the incoherent. All shared the basic mistake that they did not understand what computational universality implies about the physical world, and about human brains in particular.

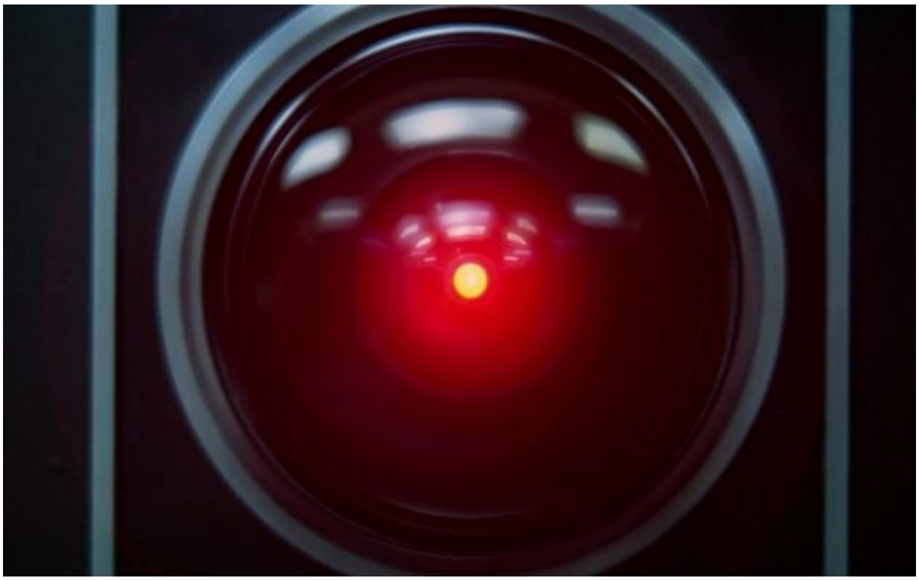
What is needed is nothing less than a breakthrough in philosophy, a theory that explains how brains create explanations

But it is the *other* camp's basic mistake that is responsible for the lack of progress. It was a failure to recognise that what distinguishes human brains from all other physical systems is qualitatively different from all other functionalities, and cannot be specified in the way that all other attributes of computer programs can be. It cannot be programmed by any of the techniques that suffice for writing any other type of program. Nor can it be achieved merely by improving their performance at tasks that they currently do perform, no matter by how much.

Why? I call the core functionality in question *creativity*: the ability to produce new explanations. For example, suppose that you want someone to write you a computer program to convert temperature measurements from Centigrade to Fahrenheit. Even the Difference Engine could have been programmed to do that. A universal computer like the Analytical Engine could achieve it in many more ways. To specify the functionality to the programmer, you might, for instance, provide a long list of all inputs that you might ever want to give it (say, all numbers from -89.2 to +57.8 in increments of 0.1) with the corresponding correct outputs, so that the program could work by looking up the answer in the list on each occasion. Alternatively, you might state an algorithm, such as 'divide by five, multiply by nine, add 32 and round to the nearest 10th'. The point is that, however the program worked, you would consider it to meet your specification — to be a bona fide temperature converter — if, and only if, it always correctly converted whatever temperature you gave it, within the stated range.

Now imagine that you require a program with a more ambitious functionality: to address some outstanding problem in theoretical physics — say the nature of Dark Matter — with a *new explanation* that is plausible and rigorous enough to meet the criteria for publication in an academic journal.

Such a program would presumably be an AGI (and then some). But how would you specify its task to computer programmers? Never mind that it's more complicated than temperature conversion: there's a much more fundamental difficulty. Suppose you were somehow to give them a list, as with the temperature-conversion program, of explanations of Dark Matter that would be acceptable outputs of the program. If the program did output one of those explanations later, that would *not* constitute meeting your requirement to generate new explanations. For none of those explanations would be new: you would already have created them yourself in order to write the specification. So, in this case, and actually in all other cases of programming genuine AGI, only an*algorithm* with the right functionality would suffice. But writing that algorithm (without first making new discoveries in physics and hiding them in the program) is exactly what you wanted the programmers to do!



I'm sorry Dave, I'm afraid I can't do that: HAL, the computer intelligence from Stanley Kubrick's *2001: A Space Odyssey*. *Courtesy MGM*

Traditionally, discussions of AGI have evaded that issue by imagining only a *test* of the program, not its specification — the traditional test having been proposed by Turing himself. It was that (human) judges be unable to detect whether the program is human or not, when interacting with it via some purely textual medium so that only its cognitive abilities would affect the outcome. But that test, being purely behavioural, gives no clue for how to meet the criterion. Nor can it be met by the technique of 'evolutionary algorithms': the Turing test cannot itself be automated without *first* knowing how to write an AGI program, since the 'judges' of a program need to have the target ability themselves. (For how I think biological evolution gave us the ability in the first place, see my book *The Beginning of Infinity*.)

And in any case, AGI cannot possibly be defined purely behaviourally. In the classic 'brain in a vat' thought experiment, the brain, when temporarily disconnected from its input and output channels, is thinking, feeling, creating explanations — it has all the cognitive attributes of an AGI. So the relevant attributes of an AGI program do not consist only of the relationships between its inputs and outputs.

The upshot is that, unlike any functionality that has ever been programmed to date, this one can be achieved neither by a specification nor a test of the outputs. What is needed is nothing less than a breakthrough in philosophy, a new epistemological theory that explains *how* brains create explanatory knowledge and hence defines, in principle, without ever running them as programs, which*algorithms* possess that functionality and which do not.

Such a theory is beyond present-day knowledge. What we do know about epistemology implies that any approach not directed towards that philosophical breakthrough must be futile. Unfortunately, what we know about epistemology is contained largely in the work of the philosopher Karl Popper and is almost universally underrated and misunderstood (even — or perhaps especially — by philosophers). For example, it is still taken for granted by almost every authority that knowledge consists of *justified*,*true beliefs* and that, therefore, an AGI's thinking must include some process during which it justifies some of its theories as true, or probable, while rejecting others as false or improbable. But an AGI programmer needs to know where the theories come from in the first place. The prevailing misconception is that by assuming that 'the future will be like the past', it can 'derive' (or 'extrapolate' or 'generalise') theories from repeated experiences by an alleged process called 'induction'. But that is impossible. I myself remember, for example, observing on thousands of consecutive occasions that on calendars the first two digits of the year were '19'. I never observed a single exception until, one day, they started being '20'. Not only was I not surprised, I fully expected that there would be an interval of 17,000 years until the next such '19', a period that neither I nor any other human being had previously experienced even once.

How could I have 'extrapolated' that there would be such a sharp departure from an unbroken pattern of experiences, and that a never-yet-observed process (the 17,000-year interval) would follow? Because it is simply not true that knowledge comes from extrapolating repeated observations. Nor is it true that 'the future is like the past', in any sense that one could detect in advance without already knowing the explanation. The future is actually unlike the past in most ways. Of course, *given* the explanation, those drastic 'changes' in the earlier pattern of 19s are straightforwardly understood as being due to an invariant underlying pattern or law. But *the explanation always comes first*. Without that, *any* continuation of *any* sequence constitutes 'the same thing happening again' under*some* explanation.

So, why is it still conventional wisdom that we get our theories by induction? For some reason, beyond the scope of this article, conventional wisdom adheres to a trope called the ‘problem of induction’, which asks: ‘How and why can induction nevertheless somehow be done, yielding justified true beliefs after all, despite being impossible and invalid respectively?’ Thanks to this trope, every disproof (such as that by Popper and David Miller back in 1988), rather than ending inductivism, simply causes the mainstream to marvel in even greater awe at the depth of the great ‘problem of induction’.

In regard to how the AGI problem is perceived, this has the catastrophic effect of simultaneously framing it as the ‘problem of induction’, and making that problem look easy, because it casts*thinking* as a process of predicting that future patterns of sensory experience will be like past ones. That looks like extrapolation — which computers already do all the time (once they are *given* a theory of what causes the data). But in reality, only a tiny component of thinking is about prediction at all, let alone prediction of our sensory experiences. We think about *the world*: not just the physical world but also worlds of abstractions such as right and wrong, beauty and ugliness, the infinite and the infinitesimal, causation, fiction, fears, and aspirations — and about thinking itself.

Now, the truth is that knowledge consists of conjectured explanations — guesses about what really is (or really should be, or might be) out there in all those worlds. Even in the hard sciences, these guesses have no foundations and don’t need justification. Why? Because genuine knowledge, though by definition it does *contain* truth, almost always contains error as well. So it is not ‘true’ in the sense studied in mathematics and logic. Thinking consists of *criticising* and correcting partially true guesses with the intention of locating and eliminating the errors and misconceptions in them, *not* generating or justifying extrapolations from sense data. And therefore, attempts to work towards creating an AGI that would do the latter are just as doomed as an attempt to bring life to Mars by praying for a Creation event to happen there.

Present-day software developers could straightforwardly program a computer to have ‘self-awareness’ if they wanted to. But it is a fairly useless ability

Currently one of the most influential versions of the ‘induction’ approach to AGI (and to the philosophy of science) is *Bayesianism,*unfairly named after the 18th-century mathematician Thomas Bayes, who was quite innocent of the mistake. The doctrine assumes that minds work by assigning probabilities to their ideas and modifying those probabilities in the light of experience as a way of choosing how to act. This is especially perverse when it comes to an AGI’s*values* — the moral and aesthetic ideas that inform its choices and intentions — for it allows only a behaviouristic model of them, in which values that are ‘rewarded’ by ‘experience’ are ‘reinforced’ and come to dominate behaviour while those that are ‘punished’ by ‘experience’ are extinguished. As I argued above, that behaviourist, input-output model is appropriate for most computer programming other than AGI, but hopeless for AGI. It is ironic that mainstream psychology has largely renounced behaviourism, which has been recognised as both inadequate and inhuman, while computer science, thanks to philosophical misconceptions such as inductivism, still intends to manufacture human-type cognition on essentially behaviourist lines.

Furthermore, despite the above-mentioned enormous variety of things that we create explanations about, our core *method* of doing so, namely Popperian conjecture and criticism, has a single, unified, logic. Hence the term 'general' in AGI. A computer program either has that yet-to-be-fully-understood logic, in which case it can perform human-type thinking *about anything*, including its own thinking and how to improve it, or it doesn't, in which case it is in no sense an AGI. Consequently, another hopeless approach to AGI is to start from existing knowledge of how to program specific tasks — such as playing chess, performing statistical analysis or searching databases — and then to try to improve those programs in the hope that this will somehow generate AGI as a side effect, as happened to Skynet in the Terminator films.

Nowadays, an accelerating stream of marvellous and useful functionalities for computers are coming into use, some of them sooner than had been foreseen even quite recently. But what is neither marvellous nor useful is the *argument* that often greets these developments, that they are reaching the frontiers of AGI. An especially severe outbreak of this occurred recently when a search engine called Watson, developed by IBM, defeated the best human player of a word-association database-searching game called Jeopardy. 'Smartest machine on Earth', the PBS documentary series*Nova* called it, and characterised its function as 'mimicking the human thought process with software.' But that is precisely what it does not do.

The thing is, playing Jeopardy — like every one of the computational functionalities at which we rightly marvel today — is firmly among the functionalities that can be specified in the standard, behaviourist way that I discussed above. No Jeopardy answer will ever be published in a journal of new discoveries. The fact that humans perform that task less well by using creativity to generate the underlying guesses is not a sign that the program has near-human cognitive abilities. The exact opposite is true, for the two methods are utterly different from the ground up. Likewise, when a computer program beats a grandmaster at chess, the two are not using even remotely similar algorithms. The grandmaster can explain why it seemed worth sacrificing the knight for strategic advantage and can write an exciting book on the subject. The program can only prove that the sacrifice does not force a checkmate, and cannot write a book because it has no clue even what the objective of a chess game is. Programming AGI is not the same sort of problem as programming Jeopardy or chess.

An AGI is qualitatively, not quantitatively, different from all other computer programs. The Skynet misconception likewise informs the hope that AGI is merely an emergent property of complexity, or that increased computer power will bring it forth (as if someone had already written an AGI program but it takes a year to utter each sentence). It is behind the notion that the unique abilities of the brain are due to its 'massive parallelism' or to its neuronal architecture, two ideas that violate computational universality. Expecting to create an AGI without first understanding in detail how it works is like expecting skyscrapers to learn to fly if we build them tall enough.

In 1950, Turing expected that by the year 2000, 'one will be able to speak of machines thinking without expecting to be contradicted.' In 1968, Arthur C. Clarke expected it by 2001. Yet today in 2012 no one is any better at programming an AGI than Turing himself would have been.

This does not surprise people in the first camp, the dwindling band of opponents of the very possibility of AGI. But for the people in the other camp (the AGI-is-imminent one) such a history of failure cries out to be explained — or, at least, to be rationalised away. And indeed, unfazed by the fact that they could never induce such rationalisations from experience as they expect their AGIs to do, they have thought of many.

The very term ‘AGI’ is an example of one. The field used to be called ‘AI’ — artificial intelligence. But ‘AI’ was gradually appropriated to describe all sorts of unrelated computer programs such as game players, search engines and chatbots, until the G for ‘general’ was added to make it possible to refer to the real thing again, but now with the implication that an AGI is just a smarter species of chatbot.

Another class of rationalisations runs along the general lines of: AGI isn’t that great anyway; existing software is already as smart or smarter, but in a non-human way, and we are too vain or too culturally biased to give it due credit. This gets some traction because it invokes the persistently popular irrationality of cultural relativism, and also the related trope that: ‘We humans pride ourselves on being the paragon of animals, but that pride is misplaced because they, too, have language, tools …

… And self-awareness.’

Remember the significance attributed to Skynet’s becoming ‘self-aware’? That’s just another philosophical misconception, sufficient in itself to block any viable approach to AGI. The fact is that present-day software developers could straightforwardly program a computer to have ‘self-awareness’ in the behavioural sense — for example, to pass the ‘mirror test’ of being able to use a mirror to infer facts about itself — if they wanted to. As far as I am aware, no one has done so, presumably because it is a fairly useless ability as well as a trivial one.

Perhaps the reason that self-awareness has its undeserved reputation for being connected with AGI is that, thanks to Kurt Gödel’s theorem and various controversies in formal logic in the 20th century, self-reference of any kind has acquired a reputation for woo-woo mystery. So has consciousness. And here we have the problem of ambiguous terminology again: the term ‘consciousness’ has a huge range of meanings. At one end of the scale there is the philosophical problem of the nature of subjective sensations (‘qualia’), which is intimately connected with the problem of AGI. At the other, ‘consciousness’ is simply what we lose when we are put under general anaesthetic. Many animals certainly have that.

AGIs will indeed be capable of self-awareness — but that is because they will be General: they will be capable of awareness of *every* kind of deep and subtle thing, including their own selves. This does not mean that apes who pass the mirror test have any hint of the attributes of ‘general intelligence’ of which AGI would be an artificial version. Indeed, Richard Byrne’s wonderful research into gorilla memes has revealed how apes are able to learn useful behaviours from each other without ever understanding what they are for: the explanation of how ape cognition works really is behaviouristic.

Ironically, that group of rationalisations (AGI has already been done/is trivial/ exists in apes/is a cultural conceit) are mirror images of arguments that originated in the AGI-is-impossible camp. For every argument of the form ‘You can’t do AGI because you’ll never be able to program the human soul, because it’s supernatural’, the AGI-is-easy camp has the rationalisation, ‘If you think that human cognition is qualitatively different from that of apes, you must believe in a supernatural soul.’

‘Anything we don’t yet know how to program is called human intelligence,’ is another such rationalisation. It is the mirror image of the argument advanced by the philosopher John Searle (from the ‘impossible’ camp), who has pointed out that before computers existed, steam engines and later telegraph systems were used as metaphors for how the human mind must work. Searle argues that the hope for AGI rests on a similarly insubstantial metaphor, namely that the mind is ‘essentially’ a computer program. But that’s not a metaphor: the universality of computation follows from the known laws of physics.

Some, such as the mathematician Roger Penrose, have suggested that the brain uses *quantum* computation, or even hyper-quantum computation relying on as-yet-unknown physics beyond quantum theory, and that this explains the failure to create AGI on existing computers. To explain why I, and most researchers in the quantum theory of computation, disagree that this is a plausible source of the human brain’s unique functionality is beyond the scope of this essay. (If you want to know more, read Litt *et al*’s 2006 paper ‘Is the Brain a Quantum Computer?’, published in the journal *Cognitive Science*.)

That AGIs are *people* has been implicit in the very concept from the outset. If there were a program that lacked even a single cognitive ability that is characteristic of people, then by definition it would not qualify as an AGI. Using*non*-cognitive attributes (such as percentage carbon content) to define personhood would, again, be racist. But the fact that *the ability to create new explanations* is the unique, morally and intellectually significant functionality of people (humans and AGIs), and that they achieve this functionality by conjecture and criticism, changes everything.

Currently, personhood is often treated symbolically rather than factually — as an honorific, a promise to *pretend* that an entity (an ape, a foetus, a corporation) is a person in order to achieve some philosophical or practical aim. This isn’t good. Never mind the terminology; change it if you like, and there are indeed reasons for treating various entities with respect, protecting them from harm and so on. All the same, the distinction between actual people, defined by that objective criterion, and other entities has enormous moral and practical significance, and is going to become vital to the functioning of a civilisation that includes AGIs.

The battle between good and evil ideas is as old as our species and will go on regardless of the hardware on which it is running

For example, the mere fact that it is not the computer but the running program that is a person, raises unsolved philosophical problems that will become practical, political controversies as soon as AGIs exist. Once an AGI program is running in a computer, to deprive it of that computer would be murder (or at least false imprisonment or slavery, as the case may be), just like depriving a human mind of its body. But unlike a human body, an AGI program can be copied into multiple computers at the touch of a button. Are those programs, while they are still executing identical steps (ie before they have become differentiated due to random choices or different experiences), the*same* person or many different people? Do they get one vote, or many? Is deleting one of them murder, or a minor assault? And if some rogue programmer, perhaps illegally, creates billions of *different*AGI people, either on one computer or on many, what happens next? They are still people, with rights. Do they all get the vote?

Furthermore, in regard to AGIs, like any other entities with creativity, we have to forget almost all existing connotations of the word ‘programming’. To treat AGIs like any other computer programs would constitute brainwashing, slavery, and tyranny. And cruelty to children, too, for ‘programming’ an already-running AGI, unlike all other programming, constitutes *education*. And it constitutes *debate*, moral as well as factual. To ignore the rights and personhood of AGIs would not only be the epitome of evil, but also a recipe for disaster: creative beings cannot be enslaved forever.

Some people are wondering whether we should welcome our new robot overlords. Some hope to learn how we can rig their programming to make them constitutionally unable to harm humans (as in Isaac Asimov’s ‘laws of robotics’), or to prevent them from acquiring the theory that the universe should be converted into paper clips (as imagined by Nick Bostrom). None of these are the real problem. It has *always* been the case that a single exceptionally creative person can be thousands of times as productive — economically, intellectually or whatever — as most people; and that such a person could do enormous harm were he to turn his powers to evil instead of good.

These phenomena have nothing to do with AGIs. The battle between good and evil *ideas* is as old as our species and will continue regardless of the hardware on which it is running. The issue is: we want the intelligences with (morally) good ideas always to defeat the evil intelligences, biological and artificial; but we are fallible, and our own conception of ‘good’ needs continual improvement. How should society be organised so as to promote that improvement? ‘Enslave *all*intelligence’ would be a catastrophically wrong answer, and ‘enslave all intelligence that doesn’t look like us’ would not be much better.

One implication is that we must stop regarding education (of humans or AGIs alike) as *instruction* — as a means of transmitting existing knowledge *unaltered*, and causing existing values to be enacted obediently. As Popper wrote (in the context of scientific discovery, but it applies equally to the programming of AGIs and the education of children): ‘there is no such thing as instruction from without … We do not discover new facts or new effects by copying them, or by inferring them inductively from observation, or by any other method of instruction by the environment. We use, rather, the method of trial and the elimination of error.’ That is to say, conjecture and criticism. Learning must be something that newly created intelligences do, and control, for themselves.

I do not highlight all these philosophical issues because I fear that AGIs will be invented before we have developed the philosophical sophistication to understand them and to integrate them into civilisation. It is for almost the opposite reason: I am convinced that the whole problem of developing AGIs is a matter of philosophy, not computer science or neurophysiology, and that the philosophical progress that is essential to their future integration is also a prerequisite for developing them in the first place.

The lack of progress in AGI is due to a severe logjam of misconceptions. Without Popperian epistemology, one cannot even begin to guess what detailed functionality must be achieved to make an AGI. And Popperian epistemology is not widely known, let alone understood well enough to be applied. Thinking of an AGI as a machine for translating experiences, rewards and punishments into ideas (or worse, just into behaviours) is like trying to cure infectious diseases by balancing bodily humours: futile because it is rooted in an archaic and wildly mistaken world view.

Without understanding that the functionality of an AGI is qualitatively different from that of any other kind of computer program, one is working in an entirely different field. If one works towards programs whose ‘thinking’ is constitutionally incapable of violating predetermined constraints, one is trying to engineer away the defining attribute of an intelligent being, of a person: namely creativity.

Clearing this logjam will not, by itself, provide the answer. Yet the answer, conceived in those terms, cannot be all that difficult. For yet another consequence of understanding that the target ability is qualitatively different is that, since humans have it and apes do not, the information for how to achieve it must be encoded in the relatively tiny number of differences between the DNA of humans and that of chimpanzees. So in one respect I can agree with the AGI-is-imminent camp: it is plausible that just a single idea stands between us and the breakthrough. But it will have to be one of the best ideas ever.