

None provided13  
essay



**ASSIGN  
BUSTER**

## ARTIFICIAL INTELLIGENCE: HOW TO GET A COMPUTER TO WRITE A TERM PAPER WITHOUT DOING ANY WORK

ALEX ZIMMERMAN, OSHKOSH NORTH HIGH SCHOOL

Can computers think? What is thinking, exactly, and how does one recognize it? What is the correlation, if any, between thinking and consciousness?

Could a computer be conscious? For years, science fiction writers have used these questions as material for their stories, from domestic robots who do all the housework to automated spaceships colonizing and mining the galaxy in the name of industry. Meanwhile, the scientific community has been slowly but steadily moving towards the point where the answers to these questions become visible. Predictions for the future in the field of artificial intelligence have traditionally been overly optimistic, but as computers become increasingly adept at simulating reason, the coming century will inevitably bring with it new ideas about what it means to be human.

The practice of ruminating on the nature of thought and consciousness is nearly as old as the human race itself. Discussions of free will and fate can be found in nearly every religion and philosophy on the face of the earth — are humans in control of their actions, or is the universe merely an exceedingly complex machine, all of whose actions are pre-determined?

Recently, the development of quantum theory has suggested what to many seems the worst alternative — that the universe is deterministically random, and nothing can be either controlled or predicted. While this line of reasoning brings to mind such glib responses as asking why criminals such be punished, if they have no control over their actions (the answer, of course, is

that judges can't control their own actions either), it also leads to a deeper understanding of the issues involved in automated reasoning and the possibilities therein. No easy definition of thought exists, and one can only experience a single version of it in a lifetime. No one can even be sure that the others he or she sees are, in fact, conscious beings. It is difficult to imagine what a computer would have to do to prove that it is conscious, or what a seemingly rational human would have to do to prove that he or she is not.

The formal study of artificial intelligence — intelligent action exhibited by non-organic objects — had its inauspicious beginnings nearly 170 years ago with Charles Babbage's conception of his Analytical Engine in 1833, an early prototype of the programmable computer (Johnson 61). Lady Ada Lovelace, upon learning of the engine, remarked that it would be possible to communicate with it as one would with a human, if only the cardboard cards that stored its instructions were punched the right way. She went on to study the capabilities of algebraic manipulation that such a machine would have, and experimented with techniques of writing programs, earning her the distinction of being popularly known as the first computer scientist. Although engineering difficulties prevented Babbage from completing the engine, it paved the way for further developments in the field of numerical computation. While the Analytical Engine was the first attempt at instantiation of what would now be known as a computer, it was by no means the first conception of one. People had long recognized the difficulties inherent in the study of mathematics and formal logical systems without any kind of automation, and used simple tools such as the abacus to remedy

them. In 1617, mathematician John Napier developed Napier's Bones, a collection of digits printed on bone that was the forerunner of the sliderule (Kurzweil 161). In what is the first reference in recorded literature to something recognizable as artificial intelligence today, a dialogue by Plato recounts Socrates as asking Euthyphro for a fail-proof algorithm to determine the nature of piety (Dreyfus 67). An algorithm, or finite collection of discrete steps leading to a determined end, is at the heart of every computer program.

Despite extensive research into the theory of computers in the late 19th and early 20th centuries, electronic programmable computers were not successfully built until the early 1940s. These computers, the most notable of which was the ENIAC, consisted of large banks of vacuum tubes and were used by the US Military to calculate missile ballistics tables. Computers were used as essentially nothing more than huge number crunchers for ten years, until they gained interest among the scientific community. In 1950, British mathematician Alan Turing published his paper "Computing Machinery and Intelligence," in which he outlines his idea for a 'Turing test' to determine whether or not a computer is intelligent. Haugeland describes the test as a game in which a person communicates with two subjects, one human and one computer. Both subjects attempt to make the tester identify them as the human. When, Turing says, the tester is able to correctly identify the human no more than half the time, the computer has won and should be thought of as intelligent (6). Turing's previous work in the 1930s had been pivotal in laying the groundwork of artificial intelligence; Turing and Alonzo Church proved independently that Turing machines, the simplest possible

computers, were capable of solving any algorithm, including theoretically all those that go on inside the human head, if given enough time and memory.

The first practical efforts at creating an intelligent program began in the 1950s as well, primarily with game playing computers. The first notable chess program, MANIAC., was completed in 1956 by Stanislaw Ulam at the scientific research center in Los Alamos that had, 15 years earlier, spearheaded the creation of the atomic bomb. According to chess master Alex Bernstein, MANIAC played a “ respectable beginner’s game” and was occasionally able to beat an opponent with little experience (qtd. in Hogan 103). In the field of checkers, Arthur Samuel developed Checkers Player as a fund-raising effort for IBM. Although Samuel admitted that he didn’t particularly enjoy checkers, his program won the Connecticut state championship before being beaten by Paaslow, the program developed at Duke University that would eventually beat the world champion (Hogan 102).

Also in 1956, Logic Theorist was created by Allen Newell, J. P. Shaw, and Herbert Simon. The program used a recursive search technique to find proofs for mathematical propositions, and was able to come up with several completely original proofs of some of the theorems in Principia Mathematica, a seminal book of mathematics. In the following year, Newell, Shaw, and Simon broadened their approach and attempted to create a program that would accomplish the same sorts of tasks as Logic Theorist in the realm of the real world. The result, the General Problem Solver, proved unable to solve any but the simplest problems (Kurzweil 199). The problems of General Problem Solver, and of many other attempts to increase the practical ability of artificial intelligence, will be discussed in the next section.

Artificial intelligence became big business in the 1960s and 1970s as development trends emphasized specialization over generalization. Programmers realized that imparting all the information a program would need to function acceptably in a real world situation was far beyond the scope of their ability, but that it was relatively easy to encode large amounts of knowledge about one specific discipline. Programs that used this method became known as expert systems, and were particularly useful in the fields of medicine and technical support. Expert systems used books of rules programmed by human experts in the subject to ask and answer simple questions in an effort to locate the cause of any given problem and propose a solution. An expert system meant to include all knowledge possessed by an average adult was begun at Stanford University in the mid-1960s, but it is still in the primary development stage 35 years later. Due to their infallibility and expendability, expert systems often proved more effective and reliable than their human counterparts within their narrow areas of knowledge. This specialization occurred in other fields of artificial intelligence as well. Huge amounts of data from thousands of widely disparate sources were collected and stored each day, and some way of picking out meaningful patterns from among the trillions of bits of information was needed. Data mining programs were the answer to this dilemma, but here too it became evident that a general purpose pattern recognition system such as the human eye would be impractical, and individual solutions were developed for applications such as financial analysis and cataloguing of astronomical objects.

This trend has continued throughout the 1990s, and nearly every computer program on the market has some amount of what would 50 years ago have

been termed artificial intelligence. Many of these have become quite competent within their specific ranges of knowledge, but the drop-off of ability once they reach the edges of this knowledge is complete and immediate. The result has been that a rising level of intelligence and seeming awareness by computer programs has become expected and is taken for granted. Several practitioners of artificial intelligence have speculated that full general intelligence will never be realized on a computer only because once a computer becomes good at something it is no longer regarded by the public as an activity that requires intelligence.

At the beginning of the new millennium, the world champion chess player is a computer; robotics and artificial vision have advanced to the point that in a few years the world ping-pong champion will, in all likelihood, be a robot; a robot has explored Mars, beaming pictures back to NASA headquarters, long before any human will ever walk on the Red Planet; an electronic paper-clip painlessly guides confused users through the operation of Microsoft Word, the state-of-the-art in word processing (and several hundred thousand times larger in terms of memory than the first word processing programs). Why is it that, throughout all of this, a sense of reason remains conspicuously absent? Will these patchwork solutions ever be fit together to provide a well-rounded intelligence? Is there anything that will remain elusively out of the grasp of computers?

### Approaches to Artificial Intelligence

When computers were first developed, it was clear that they possessed huge mathematical capabilities. Their speed and accuracy at complex calculations

had never before been seen. Many researchers in the field regarded it as just a matter of time before computers surpassed humans in the area of intelligence as well. For years, machines had far outstripped humans in strength and stamina. Now computers made it clear that humans' ability in calculation was nowhere near what could be achieved. Why should general reason and common sense, which seemed to be acquired by humans nearly effortlessly, be any different? What the initial scientists and programmers failed to realize was that the shift from the kind of logical rationalism displayed by a computer to the creative associationalism exhibited by humans is not a natural extension of similar concepts but a complete paradigm shift, and that success in one, however astonishing, does not portend success in the other.

Since the 1940s, computers have infiltrated themselves within human society, but many misconceptions continue to hold sway. Information has become the lifeblood of the modern world, just as factories were earlier and just as land was before that. A person with a computer in the early 21st century is potentially infinitely more powerful than any other person in the millions of years of human history. Still, computers were designed to be good at what humans are bad at, not at what comes naturally, and they work accordingly. A bulldozer is obviously much better than any human at some things, most notably at moving large amounts of dirt from place to place, but if one asks it what the fourth root of 1, 783 is, it won't even venture a guess. The same holds true for computers. They can answer almost (6.49812162563) instantaneously the above question, but are notoriously difficult beings with which to have an intelligent conversation.



Efforts to simulate human thought patterns by traditional means in computers are misguided. The hardware of all electronic computers today consists of the same general structure, known as the von Neumann architecture. Data and instructions are located at specific discrete addresses in memory, and are located and processed one at a time by a central processor. Variations such as parallel processing, to execute multiple instructions per cycle, or virtual memory, to store more data than the physical amount of memory allows for, are all minor changes to this basic design (Artificial Intelligence 104). To utilize this structure, programs must be clearly delineated as a set of logical, definite steps, and human reasoning processes, for the very reason that they are acquired and assimilated in the human mind so effortlessly, resist this type of delineation. (This is not to say that it could not be done; assuming no metaphysical basis for consciousness exists, it would be possible to map the structure of a brain down to the level of the elementary particle and iterate the positions of each, but as such a solution would require by today's standards a hard-drive larger than the universe and more time than that hard-drive's constituent atoms would have before deteriorating, it is probably not the optimal answer.)

A good example of this computational ineptness is the field of natural language translation. Artificial intelligence gurus in the 1950s and 1960s believed that translation would be one of the first areas to succumb to computers — what is it, anyway, but simple word replacement to account for vocabulary and word shuffling to account for grammar? Initial efforts in English-Russian translation quickly revealed that language is far more amorphous than was previously thought, often resulting in hilarity: In two

famous examples, “ The spirit is willing but the flesh is weak,” became “ The vodka is good but the meat is rotten,” and an engineering paper discussing hydraulic rams became a long discourse about water-goats (Kurzweil 406). While computers have become hundreds of times faster and programming methods have been greatly refined since these efforts, translation still has its share of difficulties. AltaVista’s online translation service, which translates to and from English, French, German, Italian, Portuguese, and Spanish, renders the preceding sentence about vodka and water-goats, translated into Spanish and back, as “ The initial efforts in the translation English-Russian revealed quickly that the language is more amorphous distant than was thought previously, often giving by result hilarity: In two famous examples, ‘ the alcohol is arranged but the meat is weak’ became ‘ the vodka is good but the meat is putrefacta’, and the hydraulicos rams discussing of paper of engineering became a long speech on water-goats.” Clearly, these programs still leave a great deal to be desired. AltaVista falls into the same trap as the original program translating ‘ spirit’ and ‘ flesh,’ and another ambiguity appears since the same word in Spanish can be used for ‘ willing’ and ‘ arranged.’ Curiously, it appears that the words ‘ rotten’ and ‘ hydraulic’ are in only the English-Spanish dictionary and not vice versa (the failure of the reverse translation in the latter case could be due to the fact that the correct spelling of ‘ hidraulicos’ is with an ‘ i’).

What is true for language translation only becomes more pronounced when the scope of a program is broadened. The General Problem Solver, mentioned above, is one such example. Given a very narrow range of options it can find an optimal solution: If it knows that it is a monkey, who

wants to eat a banana that is too high for it to reach, and that there is a chair on the other side of the room, and that it can move around, pick things up, carry them, and stand on top of them, it will succeed in getting to the banana. But a real monkey has a nearly limitless repertoire of action — it could scream at the banana, do cartwheels, stick its tail in its mouth — and yet it still carries the chair to the banana and climbs on top of it (Hogan 241).

Two techniques for remedying the problems caused by a computer's rote mechanical processes, both in a state of relative infancy, hold promise. The first, neural networking, is an attempt to mimic the actual functioning of the human brain. A neural network consists of several cells, or neurons, all interlinked. Neurons can receive inputs and fire outputs, which then affect the state of other neurons. If the sum of all the inputs a neuron receives surpasses a certain level, that neuron will fire. By tweaking the behavior of each neuron, complex problems can be solved quickly. Data, instead of being stored in discrete locations, is spread around the network, so the system is far more resilient to hardware failure — much as the human brain is often able to compensate for the loss of certain abilities if a part of it is damaged. Applications of neural networks currently include pattern recognition (optical character recognition, or scanning printed text, has reached a state of near perfection) and business management aids, especially risk assessment tools (Johnson 46).

The second technique is evolutionary program development. To develop a regular program, one or more programmers write code telling the computer exactly what to do. The programmers themselves have designed the algorithm used and understand what each part of it is meant to accomplish.

In order to develop an evolutionary program, however, the programmers start with a more or less random algorithm and a program to determine how fit that algorithm is for a certain task. The algorithm is then mutated and mated with other algorithms to produce a new generation of algorithms; the most fit algorithms of each succeeding generation are mated together. After many generations, a usable program has been developed, with code written by the computer itself. Evolutionary development has been used extensively in hardware development and chip design, and is a basic premise of artificial life, simulating primitive forms of life on a computer (Benedict 263).

Both of these approaches have shown aptitude for the kind of reasoning needed by a general artificial intelligence, and as the scientific community comes to the realization that traditional program design is prohibitive for all but the most limited of intelligences, these and other methods will be further studied, implemented, and adopted. Neural networks, evolutionary program development, and other approaches currently being developed truly represent the kind of paradigm shift needed to unite creativity and concept association with logic and order.

Several scientists have argued against artificial intelligence on both ethical and scientific grounds. Computers could never possess intelligence, such scientists say, and if they did, they could certainly never be conscious. The nature of consciousness and knowledge, as suggested earlier, is truly unknown. Be this as it may, nothing suggests that these traits are specific to organic material, or that they cannot be reproduced in silicon. The moral duties of computer scientists are more complicated. Many say that, even if the capability for artificial intelligence exists, it should not be developed. The

risk a race of intelligent computers would pose to humanity is too great to be ignored. It must be noted, however, that ethics, especially on the edges of scientific development, have rarely prevented people from doing things, and as a result of one of those developments, despotic rulers, over whom the general public holds less power than they would over an artificial intelligence, have long had the capability to annihilate life on earth.

Assuming artificial intelligence can be and is developed, what will be the ramifications for society? Several occurrences exist, known as historic singularities, which involve sufficient change and variability to make long-range prediction impossible or very nearly so. The development of language was one such singularity in human history, the discovery of fire another. Others have included the invention of the wheel and the printing press, Columbus's stumbling across America, the Industrial Revolution, and the invention of computers. Future singularities could include the full realization of human cloning, large-scale expeditions into space, contact with extraterrestrials, the development of time-travel and light-speed travel, manipulation of consciousness, mass merging of consciousness, and the end of the universe.

Artificial intelligence could be another. Current specialized intelligences and expert systems will continue to improve, making various aspects of life easier, but the existence of a full and general intelligence implies such great variability that the consequences it would have cannot be accurately predicted. An artificial intelligence of this magnitude is still years away and may not appear within the next century. In the meantime, with the growing intelligence of computer applications, the increasing automation of many

tasks, the spread and evolution of the Internet, and the increasing ease of global communication, the line between computer and human will grow vague and blurred. To one without significant computer experience, artificial intelligence will seem to be already extant. Humans have created magic countless times throughout history, and within a few years each new development is taken for granted. The gradual emergence of artificial intelligence will come as no surprise to the general public, and scientists will continue to speculate about the lack of development even as their computers are widely intelligent by the standards of ten years earlier.

---

“ AltaVista Translation Services.” Internet: \*<http://babelfish.altavista.com>\*.

Accessed 2 May 1999. Software developed by AltaVista and Systran.

Artificial Intelligence. Time-Life Books: Alexandria, Virginia, 1986.

Benedict, Michael, ed. Cyberspace: First Steps. The MIT Press: Cambridge, Massachusetts. 1991.

Dreyfus, Hubert. What Computers Still Can't Do. The MIT Press: Cambridge, Massachusetts. 1993.

Freedman, David. Brain-makers. Simon & Schuster: New York. 1994.

Forsyth, Richard, and Chris Naylor. The Hitch-Hiker's Guide to Artificial Intelligence. Chapman and Hall / Methuen: London. 1985.

Haugeland, David. Artificial Intelligence: The Very Idea. The MIT Press: Cambridge, Massachusetts. 1985.

<https://assignbuster.com/none-provided13-essay/>

Hogan, James. Mind Matters. The Ballantine Publishing Group: New York, 1997.

Johnson, George. Machinery of the Mind. Times Books: New York. 1986

Kurzweil Raymond. The Age of Intelligent Machines. The MIT Press: Cambridge, Massachusetts. 1990

Penrose, Roger. The Emperor's New Mind. Oxford University Press: Oxford, England. 1989.

Reitman, Edward. Creating Artificial Life: Self-Organization. Windcrest / McGraw-Hill: New York. 1993.

Bibliography: