❶ Can Computers Think?

The History and Status of the Debate — Map 1 of 7

An Issue Map™ Publication

Start Here

1 Alan Turing, 1950
Yes, machines can (or will be able to) think. A computational system can possess all important elements of human thinking or understanding.

"I believe that at the end of the century ... one will be able to speak of machines thinking without expecting to be contradicted."

Alan Turing

Can computers have free will?

2
Computers can't have free will. Machines only do what they have been designed or programmed to do. They lack free will, but free will is necessary for thought. Therefore, computers can't think.

free will: The ability to make voluntary, unconstrained decisions. Freely made decisions are independent of the influence of such deterministic factors as genetics (nature) and conditioning (nurture).

3
Humans also lack free will. Whether or not computers have free will is irrelevant to the issue of whether machines can think. People can think, and they don't have free will. People are just as deterministic as machines are. So machines may yet be able to think.

4 Ninian Smart, 1964
Humans are programmed. If you accept determinism, then you accept that nature has programmed you to behave in certain ways in certain contexts, even though that programming is subtler than the programming a computer receives.

5
Free will is an illusion of experience. We may think we are free, but that is just an illusion of experience. Actually, we are determined to do what we do by our underlying neural machinery.

"According to the modern scientific view, there is simply no room at all for "freedom of the human will" (1986, p. 306)." — Marvin Minsky

"[H]uman beings are slaves of brute matter, compelled to act in particular ways by virtue of biochemical and neuronal factors. What we see is the illusory nature of free will (1985, p. 109)." — Geoff Simons

6 Philip Johnson-Laird, 1988a
Free will results from a multilevel representational structure. A multilevel representational structure is capable of producing free will. The system must have levels for:
- representing options for action (e.g., go to dinner, read, take a walk);
- representing the grounds for deciding which option to take (e.g., choose the one that makes me happy, choose by flipping a coin);
- representing a method for deciding which decision-making process to follow (e.g., follow the most "rational" method, follow the fastest method).

Computers that have been programmed with such multilevel structures can exhibit free will.

Options for action / Decision-making process / Grounds for choosing a decision-making process

Philip Johnson-Laird

7 Geoff Simons, 1985
Free will is a decision-making process. Free will is a decision-making process characterized by selection of options, discrimination between clusters of data, and choice between alternatives. Because computers already make such choices, they possess free will.

8 Geoff Simons, 1985
Conditional jumps constitute free will. The ability of a system to perform conditional jumps when confronted with changing information gives it the potential to make free decisions. For example, a computer may or may not "jump" when it interprets the instruction "proceed to address 9739 if the contents of register A are less than 10." The decision making that results from this ability frees the machine from being a mere puppet of the programmer.

9 Alan Turing, 1951
Machines can exhibit free will by way of random selection. Free will can be produced in a machine that generates random values, for example, by sampling random noise.

10 Jack Copeland, 1993
Free will arises from random selection of alternatives in nil preference situations. When an otherwise deterministic system makes a random choice in a nil preference situation, that system exhibits free will. A nil preference situation is one in which an agent must choose between a variety of equally preferred alternatives (for example, whether to eat one orange or another from a bag of equally good oranges). The available alternatives may have arisen from deterministic factors, but "when the dice roll," the choice is made freely.

11
Randomization sacrifices responsibility. Machines that make decisions based on random choices have no responsibility for their actions, because it is then a matter of chance that they act one way rather than another. Because responsibility is necessary for free will, such machines lack free will.

12 A. J. Ayer, 1954
Free will is necessary for moral responsibility. Randomness and moral responsibility are incompatible. We cannot be responsible for what happens randomly any more than we can be responsible for what is predetermined. Because any adequate account of moral responsibility should be grounded in the notion of free will, randomness cannot adequately characterize free will.

13 Jack Copeland, 1993
Random choice and responsibility are compatible. An agent that chooses randomly in a nil preference situation (one in which all choices are equally preferred) is still responsible for its actions. A gunman can randomly choose to kill 1 of 5 hostages. He chooses at random, but he is still responsible for killing the person whom he picks, because he was responsible for taking the people hostage in the first place. Random choice only revokes responsibility if the choice is between alternatives of differing ethical value.

14
The helplessness argument. When agents (human or machine) make choices at random, they lack free will, because their choices are then beyond their control. As J. A. Shaffer (1968) puts it, the agent is "at the helpless mercy of these eruptions within him which control his behavior."

15 Jack Copeland, 1993
The Turing randomizer is only a tiebreaker. The helplessness argument is misleading, because it implies that random processes control all decision making—for example, the decision of whether to wait at the curb or jump out in front of an oncoming truck. All the Turing randomizer does is determine what a machine will do in those situations in which options are equally preferred.

16 Jack Copeland, 1993
Being a deterministic machine is compatible with having free will. Humans and computers are both deterministic systems, but this is compatible with their being free. Actions caused by an agent's beliefs, desires, inclinations, and so forth are free, because if those factors had been different, the agent might have acted differently.

17
Computers only exhibit the free will of their programmers. Computers can't have free will because they cannot act except as they are determined to by their designers and programmers.

18 Geoff Simons, 1985.
Some computers can program themselves. Automatic programming systems (APs) write computer programs by following some of the same heuristics that human programmers use. They specify the task that the program is to perform, choose a language to write the program in, articulate the problem area the program will be applied to, and make use of information about various programming strategies. Programs written by such APs are not written by humans, and so computers that run those programs do not just mirror the free will of humans.

19 Paul Ziff, 1959
Preprogrammed robots can't have psychological states. Because they are programmed, robots have no psychological states of their own. They may act as if they have psychological states, but only because their programmers have psychological states and have programmed the robots to act accordingly.

20 Ninian Smart, 1964
Preprogrammed humans have psychological states. If determinism is true, then humans are programmed by nature and yet have psychological states. Thus, if determinism is true, we have a counterexample to the claim that preprogrammed entities can't have psychological states.
Supported by
"Humans Are Programmed," Box 4.

de · ter · min · ism: The belief that all actions and events are determined by the influences of nature and history. Human actions result from strict causal laws that describe the brain and its relation to the world. Free will is an illusion.

21 Paul Ziff, 1959
The record player argument. A robot "plays" its behavior in the same way that a phonograph plays a record. It is just programmed to behave in certain ways. For example, "When we laugh at the joke of a robot, we are really appreciating the wit of a human programmer, and not the wit of the robot" (Putnam, 1964, p. 679).

22 Hilary Putnam, 1964
The robot learning response. A robot could be programmed to produce new behaviors by learning in the same way humans do. For example, a program that learned to tell new jokes would not simply be repeating jokes the programmer had entered into its memory, but would be inventing jokes in the same way humans do.

23 Paul Ziff, 1959
The reprogramming argument. Humans can't be reprogrammed in the arbitrary way that robots can be. For instance, a robot can be programmed to act tired no matter what its physical state is, whereas a human normally becomes tired only after some kind of exertion. The actions of the robot depend entirely on the whims of the programmer, whereas human behavior is self-determined.

24 Hilary Putnam, 1964
Reprogramming is consistent with free will. The reprogramming argument fails to show that robots lack free will for the following reasons.
- Humans can be reprogrammed without affecting their free will. For example, a criminal might be reprogrammed into a good citizen via a brain operation, but he could still make free decisions (perhaps, for example, deciding to become a criminal once again).
- Robots cannot always be arbitrarily reprogrammed in the way that the reprogramming argument suggests. For instance, if a robot is psychologically isomorphic to a human, it cannot be arbitrarily reprogrammed.
- Even if robots can be arbitrarily reprogrammed, this does not exclude them from having free will. Such a robot may still produce spontaneous and unpredictable behavior.

"That robot's been reprogrammed but it still acts spontaneously and unpredictably ..."

25 L. Jonathan Cohen, 1955
Computers do not choose their own rules. We refer to people as "having no mind of their own" when they only follow the rules or commands of others. Computers are in a similar situation. They are programmed with rules and follow commands without conscious choice. Therefore, computers lack free will.

"Put it over there." "He has no mind of his own right now. He's acting like a computer."

26 Joseph Rychlak, 1991
Computers can't do otherwise. An agent's actions are free if the agent can do otherwise than perform them. This means that an agent is free only if it can change its goals. But only dialectical reasoning allows an agent to change its goals and thereby act freely. Because machines are not capable of that kind of thinking, they are not free.
Note: Also, see the "Can physical symbol systems think dialectically?" arguments on Map 3.

27 Selmer Bringsjord, 1992
Free will yields an infinitude that finite machines can't reproduce. Unlike deterministic machines (e.g., Turing machines), persons can be in an infinite number of states in a finite period of time. That infinite capacity allows persons to make decisions that machines could never make.
Note: Bringsjord's argument is fleshed out in the "Can automata think?" arguments on Map 7. Also, see the "Can computers be persons?" arguments on this map.

Can computers have emotions?

28
Machines can't have emotions. Machines can never be in emotional states (they can never be angry, joyous, fearful, etc.). Emotions are necessary for thought. Therefore, computers can't think.

can't experience

29 Paul Ziff, 1959
The concept of feeling only applies to living organisms. Because robots are mechanistic artifacts, not organisms, they cannot have feelings.

30 J. J. C. Smart, 1964
Having feelings does not logically imply being a living organism. Although we haven't yet come across any nonliving entities with feelings, perhaps in the future we will. There is no logical contradiction in the idea of a nonliving being that has feelings.

"Rocks: nonliving and no feelings. Check! Salt: nonliving and no feelings. Check! Staplers ... check! Water ... check! Well, no nonliving things with feelings ... yet."

31 J. J. C. Smart, 1964
We can imagine artifacts that have feelings. Several cases show that artifacts could have feelings. (1) If the biblical account of creation in Genesis were true, then humans would be both living creatures and artifacts created by God. (2) We could imagine self-replicating mechanisms whose offspring would manifest small random alterations, allowing them to evolve. Such mechanisms might be considered living and at the same time artifacts.

"They are artifacts of God as well as living creatures."

32 Hilary Putnam, 1964
"Alive" is not definitionally based on structure. Because the definition of "alive" is not based on structure, it allows for nonhuman robot physiologies. Robots made up of cogs and transistors instead of neurons and blood vessels might have feelings because they might actually be alive.

33 Georges Rey, 1980
Machines lack the physiological components of emotion. Machines lack the human physiology that is essential to emotions, for example, the ability to secrete hormones and neuroregulators. Because machines can't reproduce such a physiology through abstract computational processes, they can't possess emotions.

34 Aaron Sloman, 1987
Physiology is not essential to emotion. Human emotion can be implemented on a computer because the relevant features can be modeled (the emotion's interaction with cognitive states, motivations, etc.). The physiological aspects of emotion (which include biochemistry, behavior, and proprioception) are evolutionary remnants; they are not essential.

"That's just an evolutionary remnant. It's not essential to emotion."

35 Joseph F. Rychlak, 1991
Machines can't think dialectically, and dialectical thinking is necessary for emotions. Emotions are experienced in complicated dialectical circumstances, which require the ability to make judgments about others and gauge oppositions. Machines can't reason in that way, so machines can't experience emotions.
Supported by
"Symbol Systems Cannot Think Dialectically," Map 3, Box 25.

Joseph Rychlak

36
Emotions are necessary for thought. Only systems that can be in emotional states can be said to think.

37 Geoffrey Jefferson, 1949
Emotional experience is necessary for thought. The only entities that can possess human abilities are entities that can act on the basis of felt emotions. No mechanism can feel anything. Therefore, machines can't possess human abilities, in particular, the ability to think.
Note: Also, see "Mechanisms Can't Possess Human Consciousness," Map 6, Box 10.

38 David Gelernter, 1994
Computers must be capable of emotional association to think. In order to think, a computer must be capable of a full spectrum of thought. Computers may be capable of high-end thinking, which is focused, analytic, and goal-oriented. But in order to think as humans do they must also be capable of low-end thinking, which is diffuse, analogical, and associative. For example, a flower and a flowered dress might be associated in low-end thought by a diffuse set of emotionally charged linkages.

39 Tom Stonier, 1992
Emotional machines need limbic systems. Emotional machines need the machine equivalent of the human limbic system. The limbic system subserves emotional states, fosters drives, and motivates behavior. It is also responsible for the pleasure-pain principle, which guides the activities of all higher animals. Through the development of artificial limbic systems, emotional machines will be attainable in 20–50 years.

40 Hans Moravec, 1988
Artificial minds should mimic animal evolution. The fastest progress in AI research can be made by imitating the capabilities of animals, starting near the bottom of the phylogenetic scale and working upward toward animals with more complex nervous systems.

41 Michael Scriven, 1960, *as articulated by* Arthur Danto, 1960
If a robot can honestly talk about its feelings, it has feelings. We can determine whether a robot has feelings once we configure it to (1) use English the way humans do, (2) distinguish truth from falsehood, (3) answer questions honestly. We then simply ask, "Are you conscious of your feelings?" If it says, "yes," then it has feelings.

"Are you conscious of your feelings?" "Yes."

42 Arthur Danto, 1960
The robot's dilemma. Once an advanced robot is built, the way we talk about robots, machines, and feelings will either change or will not. This poses a dilemma.

Either
English will not change,
in which case
we will be forced to say the robot is not conscious, because English speakers do not use "conscious" as a predicate for machines.

Or
English will change,
in which case
English can evolve in 1 of 2 ways.

- **Either** We simply decide to call robots "conscious," *in which case* we have an arbitrary and hence unwarranted change in the language.
- **Or** We construct a special language that applies exclusively to machines, for example, a language that uses the suffix "-m" to represent the fact that mentalistic terms like "knows" and "conscious" apply to physical events ("knows-m," "conscious-m") in machines, *in which case* words like "conscious-m" would be used for the robot in the same situations in which "conscious" would be used for humans. But a lack of knowledge about how human consciousness might correspond to robot consciousness is precisely the issue at hand.

In Either Case
No means is provided to tell whether a robot is conscious. At best the question is pushed back.

In Either Case
Simply asking the machine if it has conscious feeling will not help us determine if it does.

43 Paul Weiss, 1960
Machines cannot love or be loved. Machines, which are mere collections of parts, cannot love or be loved. Only unified wholes that govern their parts, such as humans, have the capacity to love what is lovable or be loved by those who love. Machines fail on both counts, so they are subhuman and lack minds.

44 Margaret Boden, 1977
Emotions are cognitive schemata. What is essential to emotions is the schema of cognitive evaluation that determines the relationship between the emotion and the rest of the cognitive states of the subject. In order for machines to have emotions, they must model the complex interactions involved in the use of such concepts as pride, shame, and so forth. Furthermore, these concepts must be (partially) responsible for the behavior of the system.

Margaret Boden

45 Daniel Dennett, 1978
Our intuitions about pain are incoherent. At present, it's easy to criticize the possibility of robot pain, but only because our everyday understanding of pain is incoherent and self-contradictory. For example, morphine is sometimes described as preventing the generation of pain, and sometimes as just blocking pain that already exists. But those are inconsistent descriptions. Once we have a coherent theory of pain, a robot could in principle be constructed to instantiate that theory and thereby feel pain.

"Morphine prevents the generation of pain." "No! Morphine just blocks the pain that already exists." "Once we clear up these confusions, we can implement a theory of pain on a computer."

46 Michael Dyer, 1987
Emotions can be modeled by describing their relations to other cognitive states. Modeling emotions involves two tasks: (1) the semantic task of programming a system to understand emotions, and (2) the functional/behavioral task of programming a system to behave emotionally through the interaction of emotional states and other cognitive states, such as planning, learning, and recall.

Implemented Model
47
BORIS. BORIS is a narrative reader designed to understand descriptions of the emotional states of narrative characters. BORIS can predict the emotional responses of characters and interpret those responses by tracing them back to their probable causes.

Implemented Model
48
OpEd. OpEd is an editorial reader that deals with nonnarrative editorials—for example, critical book reviews. The program tracks the beliefs of the writer as well as the beliefs the writer ascribes to his or her critics. Unlike BORIS, OpEd is able to deal with nonnarrative texts, in which "the writer explicitly supports one set of beliefs while attacking another."

Implemented Model
49
DAYDREAMER. DAYDREAMER is a stream of thought generator that specifies how representations of emotional states affect other forms of cognitive processing. It does this by concocting "daydreams" of possible outcomes and reactions and then using those daydreams to represent the stream of consciousness of the system.

50 Aaron Sloman, 1987
Emotions are the solution to a design problem. Emotions (both in organic creatures and in artificial creations) are the solution to a design problem—how to cope intelligently with a rapidly changing environment, given established goals and limited processing resources. In both humans and machines the problem is solved with intelligent computational strategies.

51 Nico Frijda and Jaap Swagerman, 1987
Emotions are manifestations of concern realization. Emotional states result from a "concern realization system" that matches internal representations against actual circumstances in order to cope with an uncertain environment. Computers that implement the concern realization system go through emotional states.

52 Andrew Ortony, G. Clure, and A. Collins, 1988
Emotions are cognitive evaluations. Emotions are determined by the structure, content, and organization of knowledge representations and the processes that operate on them. A machine equipped with the correct knowledge-handling mechanisms, which result in appropriate behavior, will have emotions.

53 Michael Arbib, 1992
Emotions color perception and action. Cognitive appraisal, in the form of knowledge representation plus appropriate behavior, is not enough to convert bare information processing into emotion. Such a theory does not account for the fact that emotions can color one's perceptions and actions. For example, the perception of a winning touchdown in a football game could be computationally modeled as knowledge representation plus appropriate behavior. But this doesn't account for the differently colored perceptions of fans of opposing teams.

54 Philip Johnson-Laird, 1988a
Feelings are information signals in a cognitive system. Feelings are needs and emotions, which correspond to information signals of two kinds: (1) needs, which arise from lower-level distributed processors that monitor certain internal aspects of the body; (2) emotions, which also arise from lower-level distributed processors but originate as cognitive interpretations of external events, especially social events. A robot could have feelings if its computational structure implemented those 2 kinds of signals.

55 Aaron Sloman and Monica Croucher, 1981
Emotions are the product of motivational representations. Emotions result from interactions between motives and other cognitive states. Motives are representations of states of the world to be achieved, prevented, and so forth. A robot with the proper motivational processes will have emotions.

56 Aaron Sloman, 1987
Hierarchical theory of affects. Emotional states arise from hierarchically structured dispositional states, that is, tendencies to behave in certain ways given certain circumstances. Higher-level dispositions influence lower-level dispositions, which in turn influence external behavior.

57 Geoff Simons, 1985
Emotion is a type of information processing. Once we understand the biochemical and cybernetic aspects of human emotion, we will be able to build computers with emotions.

58 Geoff Simons, 1985
The Turing test provides evidence for emotions as well as for intelligence. Because behavior is an important part of determining whether a system has emotions, the Turing test is useful as a test for emotional capacities as well as for general intelligence. If a robot can pass the Turing test and if it has a cognitively plausible internal structure, then it can have emotions.
Note: Also, see Map 2.

"I really do feel bad." "Yes, that's a person!"

Should we pretend computers will never be able to think?

59 *Anticipated by* Alan Turing, 1950
The heads-in-the-sand objection. The consequences of machine thought are too dreadful to accept. We should "stick our heads in the sand" and hope that machines will never be able to think or have souls.

60 Alan Turing, 1950
The transmigration consolation. The heads-in-the-sand objection is too trivial to deserve a response; consolation is more appropriate. It may be comforting to believe that souls are passed from humans to machines when humans die by the theological doctrine of the transmigration of souls.

Does God prohibit computers from thinking?

61 *Anticipated by* Alan Turing, 1950
The theological objection. Only entities with immortal souls can think. God has given souls to humans, but not to machines. Therefore, humans can think, and computers can't.

"You have no soul!"

62 Alan Turing, 1950
The theological objection is ungrounded. The view that only humans have souls is as ungrounded and arbitrary as the view that men have souls but women don't. For all we know, in creating thinking machines we may be serving God's ends by providing dwellings for souls he creates.

Can computers understand arithmetic?

63 Stanley L. Jaki, 1969; Fred Dretske, 1990
Computers can't add, much less think. Machines only operate on uninterpreted symbols. Even when they perform the operations corresponding to addition, they are merely shuffling symbols that are meaningless to them. These manipulations become mathematics only when humans interpret them.
Note: An earlier version of this claim was made by Ludwig Wittgenstein in the 1930s and published in *Remarks on the Foundations of Mathematics* (1956).

"1+1? What does that mean?" 1 + 1 = 2

64 William Rapaport, 1988
Computers can learn to add. Computers that possess internal semantic networks can learn dialectically in the same way that humans do. Thus, while they do not intrinsically know how to add, they can learn.

65 Fred Dretske, 1990
The marijuana-sniffing dog. Computers can't have an adding thought (much less have a more complex thought) because the symbols being added don't have any meaning to the computer, and they don't have any meaning because they don't play a causal role based on that meaning. A trained dog, for example, will wag its tail when it smells marijuana, but (like a robot) it's only responding because it's been trained to do so, not because the meaning of the smell causes it to wag its tail.

Is the brain a computer?

78
The biological assumption. The brain is a machine that can think. Its neurobiological processes are similar to or identical with the information processes of a computer.
Note: More specific versions of the biological assumption argument are represented on Map 3 and on Map 5.

is similar to

79 John Searle, 1992
Nothing is intrinsically a digital computer. The syntactic structures that define computers are not intrinsic to physics; they are ascribed to physical systems by humans. So the question, "Is the brain a digital computer?" is ill-defined, because syntax can be ascribed to any sufficiently complex system. Syntactic structures are not just multiply realizable in numerous physical systems, they are universally realizable in any physical system.

"That wall can be interpreted as a digital computer."

80 Jack Copeland, 1993
Programs are not universally realizable. Even if it is true that during some interval of time a pattern of molecule movements on the wall is isomorphic with, for example, the formal pattern of the WordStar computer program, the wall will not support the same counterfactuals as the program. If the WordStar program had been given different input, it would have behaved differently. But the wall, which was not engineered to implement WordStar, would not respond to different "input" (that is, a different pattern of molecular organization) in the same way. So WordStar is not universally realizable.

"That wall doesn't support the same counterfactuals as WordStar."

81 John Searle, 1992
Universal realizability is not essential to the argument. Even without universal realizability, it is still true that syntax is observer relative. And this is enough to show that nothing, including the brain, is intrinsically a digital computer.

coun · ter · fac · tu · al: A conditional (if-then) statement whose "if" clause runs counter to the facts of reality. For example, the statement, "if pigs had wings then they would fly," is a counterfactual, because the "if" clause—that pigs have wings—is false.

82
Formal programs can be realized in multiple physical media. The same formal program could be realized in a digital computer, in a human brain, in beer cans and toilet paper, or in any number of physical implementations. The program is defined solely in terms of its formal syntactic structure; its mode of physical implementation is irrelevant.
Note: For more multiple realizability arguments, see the "Can functional states generate consciousness?" arguments on Map 6 and sidebar, "Formal Systems: An Overview," on Map 7.

83
The operation of the brain is computable. Once we have a sufficient understanding of the laws of physics and the structure of the brain, we will be able to precisely simulate the operation of the brain with a computer.

84 Roger Penrose, 1990
Low-level quantum effects are uncomputable. The biological phenomena that underlie consciousness operate at a level at which quantum effects could exert an influence. Because quantum effects are not computable, the brain and consciousness may be noncomputational and nonalgorithmic.

"I'm not computable."

85 Keith Stanovich, 1990
And then a miracle happens ... Penrose does not explain how quantum effects in the brain might affect consciousness. He simply assumes that quantum effects and the brain are miraculously related.

"That's an explanation by miracle!"

86 Herbert Simon, 1995
Quantum effects are irrelevant to symbolic processes. Quantum uncertainties are unimportant to the study of symbolic thought processes, because they occur at a low level of organization and are averaged out before they can affect higher-level processes.

Can computers draw analogies?

66
Computers can't understand analogies. Computers cannot understand analogical comparisons or metaphors. For example, a machine could not understand the sentence, "She ran the like the wind."
Note: Analogy arguments are also discussed by George Lakoff in the "Symbolic Data" arguments on Map 3.

67
Computers have understood analogy. Existing models have discovered and understood analogies.

Implemented Model
68 Brian Falkenhaimer, K. Forbus, and D. Gentner, 1990
SME. SME is a structure-mapping engine that discovers analogies between domains by a set of match rules. The analogies that result are judged according to the criteria of clarity, richness, abstractness, and systematicity. SME has found mappings between heat and water flow, solar systems and atoms, and in other domains.

69 David Chalmers, Robert French, and Douglas Hofstadter, 1995
SME only draws analogies from prestructured representations. SME creates analogies using high-level representations that are structured with those specific analogies in mind. Its behavior provides no evidence of intelligence because the analogies it discovers are already built into the data it works with.
Supported by
"The Front-End Assumption Is Dubious," Box 74.

70 David Chalmers, Robert French, and Douglas Hofstadter, 1995
Objects, attributes, and relations are too rigidly distinguished by SME. In order for its analogical mappings to work, SME assumes a rigid distinction between objects, attributes, and relations. But it is unclear whether humans make such a rigid distinction. For example, we sometimes conceptualize wealth as an object that flows between people, but at other times we conceptualize wealth as an attribute that changes with each transaction we make.

71 David Chalmers, Robert French, and Douglas Hofstadter, 1995
SME's treatment of relations is too rigid. In SME, relations are treated as n-place predicates that can only be mapped to other n-place predicates. For example, attraction is a 2-place predicate that could be represented as "attracts (sun, planet)" and then mapped to "attracts (nucleus, electron)." But it is unlikely that the human mind is so rigid in its treatment of relational mappings.

Implemented Model
72 Keith Holyoak and Paul Thagard, 1989
ACME. ACME is a connectionist network that discovers cross domain analogical mappings. The ACME network uses structural, semantic, and pragmatic constraints to seek out those mappings.

73 David Chalmers, Robert French, and Douglas Hofstadter, 1995
ACME doesn't understand analogy. ACME's claim to understand analogies is overblown. All ACME does is take algebraic sentences in predicate logic notation and compare them. For example, it only understands that "Socrates is like a midwife" to the extent that it understands that "(a(b)), (c(d)) ... is similar to (A(B)), (C(D))."

"(a (b)), (c (d)) ... is similar to (A (B)), (C (D))" "That's not the same as understanding that Socrates is like a midwife."

74 David Chalmers, Robert French, and Douglas Hofstadter, 1995
The front-end assumption is dubious. Models that use preconfigured representations and hand-tailored data assume that a separate front-end module could be built that would filter sensory data into the model's representational form.

Sensory data / Front end / Computer model

75 David Chalmers, Robert French, and Douglas Hofstadter, 1995
All-encompassing representations could not be processed. The all-purpose representation that a front-end module would provide to a computer model would have to encode a vast amount of information, enough for it to adapt to all the various contexts and analogies it might be used in. Such a representation would be too bulky for efficient processing.

76 David Chalmers, Robert French, and Douglas Hofstadter, 1995
Perception depends on analogy. How we see things depends in part on what high-level analogical processes we use. For example, Saddam Hussein will be perceived quite differently depending on whether he is viewed as analogous to Adolf Hitler (a ruthless aggressor) or to Robin Hood (a generous crusader).

Implemented Model
77 Douglas Hofstadter and Melanie Mitchell, 1995
COPYCAT. COPYCAT is a model that discovers analogies using 3 components: (1) a "slipnet" of abstract Platonic concepts whose relations can change as the model runs, (2) a "workspace" of perceptual activity that acts like a short-term memory, and (3) a "coderack" of agents that are probabilistically selected to carry out tasks in the workspace. COPYCAT is neither a symbol manipulator nor a connectionist network, though it draws on both paradigms. Representations are not delivered hand-tailored to the model, but are built up through fluid interactions between low-level and high-level components.

Are computers inherently disabled?

87 *Anticipated by* Alan Turing, 1950
The argument from disabilities. Machines can never do *X*, where *X* is any of a variety of abilities that are regarded as distinctly human, for example, being friendly, having a sense of humor, making mistakes, enjoying strawberries and cream, or thinking about oneself.
Note: A great deal of the debate represented on these maps are forms of disability arguments—arguments that machines can't be creative, can't use analogies, can't be conscious, and so forth—and so could also be thought of as supports for this claim.

"You can't do *X*."

88 Alan Turing, 1950
Disability arguments derive from our limited experience with machines. Because the machines we've seen are clunky, ugly, mechanical, and so forth, we assume that a machine could never fall in love or enjoy strawberries and cream. But these are just bad inductions from a limited base of experience.

89 *Anticipated by* Alan Turing, 1950
Computers can't enjoy strawberries and cream. Computers will never possess the human ability to enjoy strawberries and cream.

90 Alan Turing, 1950
Computers may be made to enjoy strawberries and cream. Computers might be made that will enjoy strawberries and cream, but the only importance of this would be to illuminate other issues, such as the possibility of friendship between man and machine.

91 *Anticipated by* Alan Turing, 1950
Computers can't make mistakes. Computers differ from humans in that humans can make mistakes, whereas computers can't. They are easily unmasked in the Turing test, because humans would frequently make mistakes in complex arithmetic whereas computers never do.
Note: For more on the Turing test, see Map 2.

"I can't go wrong."

92 Alan Turing, 1950
Computers can make certain kinds of mistakes. Those who think computers can't make mistakes confuse errors of functioning (errors that result from the physical construction of the machine) with errors of conclusion (errors that result from the machine's reasoning process). It is true that machines can't commit errors of functioning if they are properly constructed. But machines can commit errors of conclusion, for example, by making faulty inferences based on a lack of adequate information.

"All ravens are black."

93 *Anticipated by* Alan Turing, 1950
Computers can't think about themselves. Computers cannot be the object of their own thoughts.

94 Alan Turing, 1950
Computers can be the subject of their own thoughts. When a computer solves equations, the equations can be said to be the object of its thought. Similarly, when a computer is used to predict its own behavior or to modify its own program, we can say that it is the object of its own thoughts.

95 *Anticipated by* Alan Turing, 1950
Computers can't exhibit much diversity of behavior. Humans can display much more diversity of behavior than machines ever will.

96 Alan Turing, 1950
Diversity of behavior depends only on storage capacity. Great diversity of behavior is possible for machines if they have large enough storage capacities. The objection is based on the misconception that it is not possible for a machine to have much storage capacity.

Can computers be creative?

97
Computers can never be creative. Computers only do what they are programmed to do; they have no originality or creative powers.
Note: Similar debates play out in the "Free Will" arguments.

98 *Anticipated by* Alan Turing, 1950
Machines can never take us by surprise. Machines are entirely predictable in their behavior. Because they never do anything new, they can never surprise us.

"That's not surprising at all ..."

99 Alan Turing, 1950
Computers are not entirely predictable. The belief that computers are entirely predictable arises from the false assumption (widespread in philosophy and in mathematics) that humans can know everything that follows deductively from a set of premises. But humans learn new things in part through the working out of deductive consequences. Similarly, humans don't know everything a computer will do given some initial state of the computer; we learn new things in part by watching them perform their calculations.

100 Alan Turing, 1950
Machines frequently take us by surprise. Computer users and even experts are often surprised by the things that computers do.

"Wow! What a surprise!"

101 *Anticipated by* Alan Turing, 1950
Surprise is a result of human creativity. Even if we are surprised by what a machine does, that reaction does not mean that the machine has done anything original or creative. It just means that the human made a creative prediction about what the computer would do, and was then surprised when the computer acted differently.

"The machine isn't creative, but the human is creatively surprised."

102 Alan Turing, 1950
The argument from human creativity applies to any case of surprise. You could always say that being surprised came from you, the interpreter, rather than from anything original on the other person's or machine's part. For example, if a human surprises you with a joke, then you could argue that the surprise was a result of your interpretation of the joke rather than anything creative on the joke teller's part.

103 Countess of Lovelace, 1842
The analytical engine can never do anything original. The analytical engine (see sidebar, "The Analytical Engine") could never discover any new facts. It is limited to drawing out consequences of facts that it has been provided with. In contemporary terms, a computer can only do what it has been programmed to do.

"The analytical engine has no pretensions to originate anything. ... It can follow analysis; but it has no power of anticipating any analytical relations or truths."

PHOTO # 32
Refer to photocopy for precise cropping instructions.
Countess of Lovelace

104 Alan Turing, 1950
The analytical engine may have been able to think for itself. Ada Lovelace was justified in denying that the analytical engine could be creative, because she had no evidence that it was creative. But because the analytical engine was in fact a universal digital computer, it may have had far greater capabilities than she realized. With added speed and storage capacity the analytical engine may have been able to think for itself.

"It can do more than you realize, Ada."

The Analytical Engine
Invented by Charles Babbage circa 1860, the analytical engine was a mechanical computer composed of gears, cranks, and wheels, which could be programmed by punch cards. In principle, Babbage's analytical engine could carry out any of the calculations a modern electronic computer can, but due to construction and design costs the analytical engine was never built during Babbage's lifetime (several have been constructed since).

Charles Babbage (Corbis-Bettman)

105
Computers have already been creative. Computer models that exhibit creativity or at least some component of creativity have already been developed.

106 Douglas Hofstadter, 1995
The ELIZA effect. The ELIZA effect is a tendency to read more into computer performance than is warranted by their underlying code. For example, the computerized psychotherapy program ELIZA (see "ELIZA," Map 2, Box 34) gives apparently sympathetic responses to human concerns, but in fact is only utilizing a set of canned responses.
Note: The ELIZA effect was recognized and described by ELIZA's creator, Joseph Weizenbaum, though he didn't give it that title.

Douglas Hofstadter

Implemented Model
107 H. Gelernter, 1963
The geometry program. The geometry program is a system that works backward from geometric theorems, searching for their proofs by means–end analysis. This planning breaks down the problems using a hierarchy of goals and subgoals. To avoid impossible searches the program uses heuristics to select the most promising search paths.

Implemented Model
108 Philip Johnson-Laird, 1988a
The jazz generator. The jazz generator produces chord sequences and uses them to improvise chords, bass-line melodies, and rhythms.

Implemented Model
109 Margaret Masterman, 1971
Haiku program. A program has been written that develops haiku (a style of Japanese poetry) through interaction with humans. The model provides poets with synonym lists to aid in word choice and also constrains line length to ensure that the haiku is properly formed. The haiku program can run without human interaction by making arbitrary choices from its synonym lists.

"All white in the buds / I flash snow peaks / in the spring / Bang the sun has fogged."

Implemented Model
110 Jim Meehan, 1975
TALE-SPIN. This program writes stories with characters that have goals and subgoals dependent on their motivations. Its characters cooperate in each other's plans and can form competitive relationships when necessary to achieve their goals. The program can also represent a wide range of communications between its characters.

"... George Ant was very thirsty. George wanted to get some water. George walked from his patch of ground across the meadow through the valley to a river bank ..."

Implemented Model
111 Harold Cohen, B. Cohen, and P. Nii, 1984
AARON. AARON produces visual art by selecting a random starting point on a canvas and then drawing lines from that point using a complex set of if-then rules.

Implemented Model
112 Margaret Boden, 1990
Connectionist systems exhibit creativity. Connectionist networks can learn to recognize patterns without being specifically programmed to do so.
Note: Also, see Map 4.

"The computer recognizes that letter *A* without having been programmed to do so."

Implemented Model
113 Sheldon Klein, 1975
Book generator. This automatic novel writer generates 2,100-word mysteries. It develops a rudimentary plot based on the conflicting motivations of its characters and fits the model of a mystery story by revealing the murderer at the end.

"... Lady Buxley was near James. James caressed Lady Buxley with passion. James was Lady Buxley's lover ..."

114 Margaret Boden, 1977
The book generator is inadequate. The book-writing program's fiction is inadequate for the following reasons. (1) The stories are shapeless and rambling. (2) The specific motivational patterns are relatively crude and unstructured. (3) The identification of the murderer comes as a statement rather than as a discovery.

Can computers reason scientifically?

115
Computers can't reason scientifically. Computers are unable to think and reason as human scientists do.

116 Harry Collins, 1994
Scientific reasoning requires social agreement. Computers cannot reason scientifically because they are not members of society. Scientific laws and data do not follow from the application of an algorithm, but are developed through a quasipolitical process of negotiation.

Harry Collins

117 Harry Collins, 1994
The socialization test. The importance of socialization is demonstrated by the "socialization test," a variant of the Turing test. In the socialization test, a human control and a machine are both given a passage of "mucked-up" English. Both the machine and the human control must correct all the errors and transliterate the passage into normal English. If a judge cannot tell which text was error-corrected by machine and which by the human control subject, then the machine passes this test for socialization.
Note: For more on the Turing test, see Map 2.

"I throwed trash the wastebasket." → "I threw the trash in the wastebasket."

118 Carl Hempel, 1985
Computers can't introduce new terms or explanatory principles. A computer cannot be original because it cannot introduce new theoretical terms or principles. Computers' "discoveries" are limited to those that can be expressed using the program's fixed vocabulary and conceptual apparatus. Human discovery, by contrast, involves the introduction of new terms and principles that cannot be defined in terms of those previously available.

"It's vocabulary is fixed! It can't come up with any new terms or principles ..."

119 Richard Scheines, 1988
Computers can introduce new terms. Computers can introduce new terms using automated principles of explanatory adequacy. This has been shown using a program that uses explanatory adequacy principles to introduce new terms in the domain of "causal models"—a class of mathematical theories popular in social science.

120 Carl Hempel, 1985
Computers can't adequately evaluate hypotheses. A computer model of scientific discovery would have to use a criterion of preference to choose between hypotheses that account for available data equally well. But criteria of preference tend to be imprecise and idiosyncratic, so it is unlikely that such a criterion could be implemented on a computer.

121
Computers have already reasoned scientifically. Computer systems exist that have reasoned as scientists do, proposing explanatory hypotheses and choosing among them.

Implemented Model
122 Pat Langley, Hubert Simon, Gary Bradshaw, and Jan Zytkow, 1987
BACON. A program for discovering laws from data by applying heuristics, BACON has discovered Kepler's law of planetary motion, Galileo's law of uniform acceleration, and Ohm's law of electrical resistance.
Note: The history of BACON program is compex and extends back into the 1960s.

123 Harry Collins, 1994
BACON only works when humans filter its data. Bacon only works through its interaction with scientists who filter its data and thereby predetermine its results. If humans did not constrain its data, it is doubtful that BACON would produce any original science.
Supported by
"The Front-End Assumption Is Dubious," Box 75.

Implemented Model
124 B. G. Buchanan, D. H. Smith, W. C. White, R. Gritter, E. A. Feigenbaum, J. Ledergerg, and C. Djerassi, 1976
DENDRAL. DENDRAL is an expert system that analyzes and identifies chemical compounds by forming and testing hypotheses from experimental data. Meta-DENDRAL, a component of DENDRAL, has discovered how to synthesize previously unknown chemical compounds as well as entirely new rules of chemical analysis. It even has a publication to its credit.

Can computers be persons?

125
Computers can't be persons. Machines can never be persons. They lack ethical status and cannot bear responsibility for their actions. At best they can display personlike behavior.
Note: Many other arguments about computers being persons permeate the maps but have been placed in other regions to emphasize what specific aspect of machinehood or personhood is in question.

126 John Pollock, 1989
An artificial person can be built. An artificial person can be built from physical ingredients provided it adequately models human rationality, which is the suitable structure necessary for personhood.

127 Selmer Bringsjord, 1992
Robots can do intelligent things but will never be persons. AI will eventually succeed in building robots that can behave intelligently but will never make robots that are actually persons. Persons are genuine things (rather than logical constructions) that bear psychological properties and that can bring about states of affairs in the world.
Note: Bringsjord supports his claim with a wide range of arguments that are dispersed throughout the maps. See the "Can computers have free will?" arguments on this map, and the "Can automata think?" arguments on Map 7.

Personhood: Historical Background
Many contemporary and historical debates have dealt with the concept of personhood. The abortion debate deals with the status of the fetus as a person. Animal rights theorists ask whether various species of animals are persons or not. The emancipation of the slaves was won when the Supreme Court was convinced that African Americans were people and not property.

The question of whether robots are persons has been asked since at least the release of Karel Capek's play *R.U.R.* (Rossum's Universal Robots) in the 1920s. This play—from which the name "robot" derives—is about the struggle of intelligent robots to gain their civil liberties.

In the debate over artificial intelligence, personhood again becomes an issue, because if computers are able to think, then their ethical status may have to be upgraded. Moreover, many artificial intelligence researchers hold the dream of creating artificial life in the form of an artificial person, in part because the concept of intelligence is closely related to the concept of personhood. Some think that a thinking computer would be straight-off a person, because we know that thinking is (to some degree) part of being a person. Some think that a robot cannot think unless it is a genuine person, because otherwise there would be no "one" doing any thinking.

128 Dwight Van De Vate Jr., 1971
A machine isn't a person unless society deems it one. A machine or an individual is not a person until society collectively declares it one. This requires having a gender, a flesh-and-blood body, the ability to feel pain, and so forth. If a machine lacks any of these—if, for example, it is disembodied and can't feel pain—it won't be recognized as or treated as a person.

"That ain't a person like us!" "That's just some crummy machine ..."

129 *Anticipated by* Dwight Van De Vate Jr., 1971
Machines can behave like persons in the imitation game. A machine could treat others like a person and be treated like a person in an imitation game.
Note: Also, see the "Can the imitation game determine whether computers can think?" arguments on Map 2.

"You see? I'm a human just like you! I won the game, I fooled you every time, I can do anything a human ..."

130 Dwight Van De Vate Jr., 1971
Laboratory performance isn't enough for full reciprocity of social behavior. A machine in a lab playing the imitation game is not yet a person because it is not really being treated like one. It's treated like an artifact in an experiment, which we can unplug and ignore as we see fit.

"Let's go to lunch." "..."

131 Dwight Van De Vate Jr., 1971
Reciprocity of social behavior is required for personhood. Persons must:
- be capable of treating others like persons in a variety of contexts;
- be treated like a person by members of society in a variety of contexts.

you so I treat you like me treat you so me treat I like

One of 7 in this Issue Mapping™ series—Get the rest!

Legend

The arguments on these maps are organized by links that carry a range of meanings:

is supported by — Arguments that uphold or defend another claim. Examples include: supporting evidence, further argumentation, thought experiments, extensions or qualifications, and implemented models.

is disputed by — A charge made against another claim. Examples include: logical negations, counterexamples, attacks on an argument's emphasis, potential dangers an argument might raise, thought experiments, and implemented models.

is interpreted as — A distinctive reconfiguration of an earlier claim.

Unmapped Territory (Additional arguments) — This icon indicates areas of argument that lie on or near the boundaries of the central issue areas mapped on these maps. It marks regions of potential interest for future mapmakers and explorers.

Focus Box: The lowest-numbered box in each issue area is an introductory focus box. The focus box introduces and summarizes the core dispute of each issue area, sometimes as an assumption and sometimes as a general claim with no particular author.

Arguments With No Authors: Arguments that are not attributable to a particular source (e.g., general philosophical positions, broad concepts, common tests in artificial intelligence) are listed with no accompanying author.

Citations: Complete bibliographic citations can be found in the booklet that accompanies this map.

Methodology: A further discussion of argumentation analysis methodology can be found in the booklet that accompanies this map.

Anticipated by — Where this phrase appears in a box, it identifies a potential attack on a previous argument that is raised by the author so that it can be disputed.

As articulated by — Where this phrase appears in a box, it identifies a reformulation of another author's argument. The reformulation is different enough from the original author's wording to warrant the use of the tag. This phrase is also used when the original argument is impossible to locate other than in its articulation by a later author (e.g., word of mouth), or to denote a general philosophical position that is given a special articulation by a particular author.

Version 1.0

The Issue Mapping™ series is published by MacroVU Press, a division of MacroVU, Inc. MacroVU is a registered trademark of MacroVU, Inc. Issue Map and Issue Mapping are trademarks of Robert E. Horn.