Turing's Imitation Game: a discussion with the benefit of hind-sight

Kirkpatrick, B. and Klingner, B.

This paper was prepared after the October 19, 2004 discussion in the Berkeley Computer Science course "Reading the Classics" with professor Christos H. Papadimitriou.

***Abstract*— Alan Turing proposed the imitation game in his 1950 paper titled "Computing Machinery and Intelligence" [1]. This paper discussed the arguments for and against artificial intelligence that existed then. Since this time, science has advanced in areas of neuroscience, machine learning, and computer theory granting us a different perspective on the argument. This paper discusses the history of Artificial Intelligence (AI), progress made in the field, and the arguments against AI. It also summarizes the discussion from the 19 October meeting of "Reading the Classics."**

***Index Terms*— Alan Turing, Imitation Game, Chinese Room, Artificial Intelligence**

I. The Imitation Game

IN the world of artificial intelligence, it is tantalizing to ask the question, "Can machines think?" The question is problematic, though. What does it mean to think? Could it be that by its very definition, thinking is something that only humans can do? Turing conceived of the "imitation game" as a way to side-step this question, and define a metric of success for artificial intelligence researchers that has still not been met.

The initial version of the imitation game, as put forth by Turing, involves three humans. A man, referred to as "A," a woman, referred to as "B," and an interrogator, referred to as "C." C is allowed to communicate with A and B only through a terminal (i.e., by typing text questions and receiving answers in text). C's goal is to determine whether A or B is the real woman. A and B each want to convince C that he/she is the real woman.

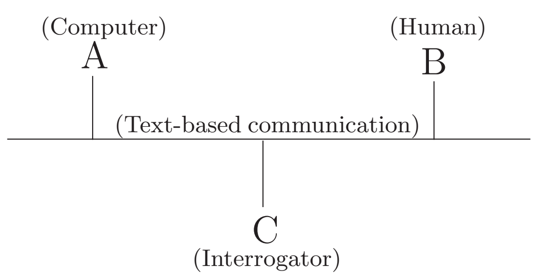

Fig. 1. **The Imitation Game:** The goal of the interrogator is to determine which of the two participants (A or B) is the machine and which is the human. There are physical boundries between the participants, and the communication is accomplished via text.

Turing then modifies the imitation game, replacing A with a machine. C's goal now becomes to pick out which of A and B is human [Fig. 1]. Basically, the machine has to fool a human interrogator into believing that it is human [1]. Framing the test in this way sets a high bar for the computer, but totally avoids the question, "can computers think?" The class seemed to accept this test without too much debate. One student brought up Searle's view that this test only verifies "weak" AI and not "strong" AI. This distinction will be addressed in a later section.

To have a little fun, we conducted our own in-class Turing Test, where we displayed five stanzas of poetry, some written by a computer, and some written by humans. The class was asked to differentiate between the two, as a demonstration of a field in which computers can convincingly imitate humans. Not many people guessed correctly on all five, which is not surprising given the length of the excerpts and their lack of context. In fact, when experienced human judges were given thirty such stanzas, they correctly identified the computer output only about half of the time [2].

II. History of Artificial Intelligence

The idea of a machine that can imitate a human is a very old one that dates at least from the time of Leonardo da Vinci in 1495 when he drafted one of the first robot designs. His design included elbow joints that were modeled after the human elbow. Since then the idea of thinking machines has thrived in the world of fiction. One of the more notable fictional developments of robot characters is in Issac Asimov's books. He was responsible in 1940 for conceiving of the "Three Laws of Robotics," which were designed to make robots subservient to humans.

In the real world, the idea of Artificial Intelligence began to capture the imagination of computer scientists and mathematicians in the 1940s and 50s. The idea of modeling neurons with algorithms was first put forth by McCulloch and Pitts in 1943 [3]. In 1950, Alan Turing proposed the imitation game in his paper titled "Computing Machinery and Intelligence" [1]. He was joined in his optimism and interest in AI by many contemporaries. In the same year, Claude Shannon described an algorithm for playing chess. Alan Newell and Herbert

Simon developed one of the first expert systems in 1956 and named it the Logic Theorist. Noam Chomsky was doing parallel work into trying to analyze language mathematically, and he published *Syntactic Structures* in 1957 [4]. Genetic algorithms were proposed by Friedberg in 1958 [5].

A well known computer program named ELIZA by Joseph Weizenbaum was somewhat successful at the imitation game in 1966. This program operated a bit like a human psychologist and convinced people to talk to it. Research into neural networks continued into the 1960s when Minsky and Papert slowed research in the area with their book *Perceptrons* which explained the limits of single-layer neural networks [6]. In 1976 the introduction of the Kurzweil Reading Machine allowed blind people to hear the machine read printed text. The 1979 MYCIN expert system was able to diagnose some diseases at least as well as doctors.

During this entire time, the proponents of the AI field have continuously made ambitious predictions about the success of AI, and opponents have criticized the goals and ideas of the field. Herbert Simon, Nobel Prize laureate, 1957 predicted in 1958 that “within ten years a digital computer will be the world’s chess champion” [7]. In 1972, Hubert Dreyfus fiercely criticized the goal of artificial intelligence in his book *What Computers Can’t Do* by claiming that human intelligence is based on more than just the manipulation of symbols [8]. John Searle raised another objection in 1980 to the strong AI idea that machines can think when he published the Chinese Room thought experiment [9]. These objections to AI have all been raised in the area of philosophy, and neither this paper’s authors nor the class knew of any mathematical objections to AI.

III. AI Successes

It’s worthwhile to review a few of the major perceived successes of artificial intelligence since Turing wrote his paper in 1950. A few that might seem significant to laypeople and also to computer scientists are addressed below.

A. Chess

Computer scientists have long been fascinated by problem of teaching a machine to play chess. Many algorithms and machines have been put forth over the years, increasing in complexity and sophistication. Computerized chess play reached what most believe is its ultimate culmination when, in 1997, Deep Blue beat the world chess champion Gary Kasparov. Discussion on this point brought up several interesting facets of the success: Some people mentioned that Gary lost only because he was unprepared for Deep Blue’s style of play. They argued that it might have been unfair since Deep Blue had been ‘primed’ using all of Gary’s past games, while Gary was not allowed to review the way in which the machine worked. Another student pointed out that what Deep Blue does is not AI in the sense that most current researchers would acknowledge it; rather, it is just a tree searching algorithm combined with heuristics to value different board positions.

Despite these objections, beating the world chess champion was a major step forward for machine-based game playing, and no doubt Turing would have regarded it as a step toward passing his imitation game. Also, Deep Blue would probably be first to come up in a lay-person’s review of great successes of computer intelligence.

B. Speech Recognition

One aspect of human intelligence that seems unique among animals is the capability to understand speech. Almost all humans, as part of their normal development, learn to understand and speak a language to communicate with other humans. It’s natural to expect that a “thinking” computer would also have to understand human speech in some form.

Great strides have been made over the last couple of decades in speech recognition software development. Ray Kurzweil in particular has been responsible for developing increasingly sophisticated continuous speech recognition systems. Today, a number of commercial products from companies like IBM offer people the ability to speak to a computer for dictation or control. The sense that a computer “understands” when you talk to it is a compelling sensation that the computer can actually “think.”

In truth, though, modern speech recognition has nothing to do with understanding. As was brought up in the discussion, most current algorithms are based on statistical analysis, training, and built-in dictionaries. They require training, and there is certainly not a speech recognition program that can understand any language spoken to it. Additionally, although these algorithms are capable of using context to pick out words, they don’t know what any of the words mean. As one student mentioned, they are basically signal processing applications specialized to reconstruct language.

C. Leobner Prize

It’s worth mentioning that people haven’t given up on trying to pass the Turing test. In fact, an annual competition is held in the hope that someday someone will pass the test. As described on its website:

> In 1990 Hugh Loebner agreed with The Cambridge Center for Behavioral Studies to underwrite a contest designed to implement the Turing Test. Dr. Loebner pledged a Grand Prize of $100,000 and a Gold Medal for the first computer whose responses were indistinguishable from a human’s. Each year an annual prize of $2000 and a bronze medal is awarded to the most human computer. The winner of the annual contest is the best entry relative to the other entries that year, irrespective of how good it is in an absolute sense.

Some in the class remarked that the computers had great success in imitating dysfunctional humans. It is easy for a program to function like an autistic person than a more normal person.

D. Computer Vision

Another area of computer science research that produces results that look like “thinking” is computer vision. Vision systems are now capable of recognizing faces in a crowd, reading and sorting mail, and performing complex compositing operations. The ability for a computer to see and understand its world is another example of a human trait being expressed by a machine.

Discussion on this point brought up the obvious objections, similar to those regarding speech recognition. The processes involving computer vision are mostly statistical, and reflect no real understanding of what is being looked it. Still, the techniques applied are more closely allied to those of AI researchers than those of speech recognition.

IV. Objections to the Turing Test

In this section, we discussed some of the objections to AI that Turing argued against. In addition, we briefly discussed objections that have been raised since then by people such as Searle's Chinese Room argument against strong AI and Dreyfus's 1972 argument regarding the non-symbolic basis for human intelligence.

A. Head-in-the-sand

When a person refuses to consider that a machine could pass the Turing test, then likely they are likely using the head-in-the-sand argument. Typically people who argue this way think that AI raises too many questions about the nature of thought for them to consider the idea, so they ignore it. This idea is behind many of the fictional accounts of the future. There is a plethora of movies and books that invoke fear in readers when they describe thinking machines that take over the world or commit other atrocities.

B. Mathematical Limitations

Both Godel's incompleteness theorem and the halting problem could be raised as objections to Artificial Intelligence. In addition to these two objections that Turing discussed in his paper, we will also discuss another argument that attempts to bring up the complexity of natural language.

Godel's incompleteness theorem states that for any consistent logical system that includes number theory, there exist statements that cannot be proved or disproved within that system. His second theorem shows that no consistent logical system can prove its own consistency. The halting problem proved by Turing showed that no machine can determine whether another machine will halt when processing a given input. These results indicate the limits of the mathematical theory that we use to understand computers. But, as Turing observed, they do not necessarily pose a problem for the machine playing the imitation game. Humans are fallible and often do not know the answers to questions. A human judging the imitation game may have a difficult time determining whether an inaccurate or nonexistent response is due to human error or the mathematical limitations of the machine.

The possible computational complexity of natural language is another source of concern to AI. The majority of Noam Chomksy's work in the area of language classification was done after Turing's death, and these ideas were not addressed in "Computing Machinery and Intelligence." Chomsky's language hierarchy includes regular languages, context-free languages, context-sensitive languages, and recursively enumerable languages which increase in expressive power from regular to recursively enumerable languages. Most notably, natural languages do not appear in this hierarchy, because they are ambiguous and difficult to describe as mathematical objects. This fact could be raised as an argument against computers being able to convincingly imitate human language. Yet, the class decided that this argument does not delineate an actual limitation to computers. Because natural languages are not mathematical, there are no theorems about the computability of natural language. In addition, machine learning techniques, which are statistical algorithms, have made progress in the area of natural language processing. Today, both machine learning and knowledge representation methods are areas of intense study.

Several people in class remarked that a mathematical limitation does not necessarily translate into a practical limitation. Checkers and chess are both EXPTIME-complete in their generalized $n*n$ board version [10], [11], but in practice, both checkers and chess programs can beat the best human players. With powerful enough computers and clever algorithms, theoretical limitations can be overcome in practice.

C. Consciousness

As Turing summarizes this objection with a quote from Jefferson Lister, "Not until a machine can write a sonnet or compose a concerto because of thoughts and emotions felt, and not by the chance fall of symbols, could we agree that machine equals brain-that is, not only write it but know that it had written it. No mechanism could feel (and not merely artificially signal, an easy contrivance) pleasure at its successes, grief when its valves fuse, be warmed by flattery, be made miserable by its mistakes, be charmed by sex, be angry or depressed when it cannot get what it wants."

Turing points out that this argument is really just an attack on the validity of the Turing test. But Turing has, at this point, already gone to great pains to address the fact that his test is not a test of whether a computer can think or feel, but only a test to see whether the interrogator can be fooled.

This objection was refined in a thought experiment called "The Chinese Room" by Searle. Searle describes a room in which a man who speaks only English sits with a book that has instructions for, given Chinese characters as input, construct an appropriate response. A man fluent in Chinese outside the room sends in messages written in Chinese, and the man inside responds. The fluent man cannot distinguish between the man and the book and a true Chinese speaker. Searle argues that this shows that nothing in the box "knows" Chinese, and yet the test is still passed. Again, we see that this is not an objection in itself, but rather another question of the validity of the test. Searle claims that AI like that demonstrated by the Turing test is "weak" AI, whereas a machine that actually knew Chinese would be considered to have attained "strong" AI.

The discussion this raised brought up a valid rebuttal of Searle's argument: he adds nothing nor takes nothing away by making the distinction between strong and weak AI. He merely renames the question “can machines think” to “can machines possess strong AI,” without tackling the prickly task of defining what thinking is.

D. Creativity/Surprise

Lady Lovelace put forward an objection to the Turing test, claiming that a computer cannot originate anything. Turing first points out that it’s hard to pin down the originality of anything, whether it is produced by a computer or by a human, or whether that which seems original is merely emergent from some teaching or environmental factor.

Beyond this, Turing mentions rightly that anyone who has used a computer knows that, though every part of it is designed by humans, the humans cannot instantaneously understand all of the workings of the machine, and can be surprised by what it does. Discussion in class confirmed that anyone who has every tried to program a modern computer has experienced behavior that they did not intend to provoke.

E. Continuity of Nervous System and Other Biological Arguments

It is worth remarking that the nervous system is not a discrete state machine. Turing noted this as a source of arguments opposed to his own view. While it is true that the brain is not a discrete state machine, neuroscience has made much progress in cracking the neural code. Multiple spike train data is being collected and analyzed mathematically [12]. Neuroscientists are very hopeful that these data will provide the key to cracking the neural code and to simulating the functioning of the brain.

Some people were inclined to argue for Artificial Intelligence based on the discreteness of DNA. The genome of an organism is, in some sense, a discrete program that determines how an organism will look, function, and interact with its environment. But as the class remarked, the environment in which the DNA operates is not discrete. The physics that operate on the molecules in an organism is not discrete. This is clearly seen in our current knowledge of proteins which account for many of the functions of the DNA “program”. Our current model of protein folding which considers the physics of the molecules is known to be NP-complete. Christos brought out that fact that a model is different from reality, and better models and better algorithms could be discovered in the future. Hence, arguments based on discrete DNA or the difficulty of some biological modeling problems are not clearly in favor of or in opposition to the goal of achieving artificial intelligence.

F. Learning Machines

The objection to the Turing test which seems least well-handled by Turing is that of learning machines. It is in fact his only positive argument for a machine passing the test, while all the other bits are only refutations of possible objections that could be raised to it. Turing posits “Presumably the child brain is something like a notebook as one buys it from the stationer’s. Rather little mechanism, and lots of blank sheets.” He goes on to describe a system by which a replica of a child brain could be constructed, and then information fed to it until it resembled that of an adult.

The most obvious problem with this approach is that it is not clear at all that the child mind has the form which he describes. Indeed this bring us back to Dreyfus’s argument that there is not a symbolic basis to human intelligence. This idea is known as the antithesis to the strong Whorfian hypothesis in the linguistics community. There is evidence that there is a non-symbolic basis to numerical thoughts [13]. Indeed, as some in the class pointed out, this idea of the Whorfian hypothesis is hotly contested in linguistic circles, and their are strong arguments both for and against the idea that language determines thought.

Turing’s argument also appeals to the growth rate of digital memories, where he posits that the human brain has on the order of a billion decimal digits of storage. By modern standards this seems like a gross underestimate. Also, he claims that once the capacity is available, the problem of constructing a computer system of the form of a human mind is “mainly one of programming” [1]. We now find ourselves in a world with almost limitless storage capacity, but what we lack is any sense of how to use that memory to mimic the human mind.

V. Thinking Computers

Turing makes a bold assertion in his paper: "I believe that at the end of the century the use of words and general educated opinion will have altered so much that one will be able to speak of machines thinking without expecting to be contradicted.” We have now past the end of the century. Has Turing’s prediction come to pass?

At first blush, a computer scientist would say no. But consider the following exchange between an interviewer and Gary Kasparov’s technical advisor after Gary lost the historic chess match to Deep Blue:

> MARGARET WARNER: All right. Let me bring Mr. Friedel back in here. Mr. Friedel, did Gary Kasparov think the computer was thinking?
>
> FREDERIC FRIEDEL: Not thinking but that it was showing intelligent behavior. When Gary Kasparov plays against the computer, he has the feeling that it is forming plans; it understands strategy; it’s trying to trick him; it’s blocking his ideas, and then to tell him, now, this has nothing to do with intelligence, it’s just number crunching, seems very semantic to him. [Friedel is Kasparov’s technical advisor.] [14].

One could argue that Gary Kasparov is a very smart person, and it seems that to him the question of whether or not the computer was thinking is only a “semantic” distinction. Also, consider the casualness with which people today say things like “Word thinks I’m writing a letter,” “Windows is unhappy,” et cetera. These casual statements could belie an implicit assumption that a computer is thinking or feeling.

An important objection to this was brought up in class discussion: long before computers entered mainstream use, people were accustomed to ascribing similar emotions and thoughts to simpler appliances like refrigerators, animals, and even to the weather, as in "the clouds look angry." Perhaps our anthropomorphism of computers is just an extension of this inclination.

In addition, one can ask the question "can ships swim? [15]" This question brings out the differences in nomenclature that exist to indicate shades of differences between meanings. Most of us would admit that ships do not swim like humans, but they do accomplish the feats of floating and moving through the water. As Christos pointed out, perhaps this idea is illuminating when discussing whether or not computers think.

VI. If the Test were Passed...

We ended the discussion with a quick mention of questions that we might one day face, if a computer does ever manage to pass the Turing test:

Does the machine have rights?

Can the machine commit crimes?

Who pays the power bill?

What if the machine reproduces?

These questions seem academic, and have been the subject of many science fiction books and movies. But some in the class believe that they are serious questions, and Christos stated that he thought we would be forced to answer these questions "in our lifetimes."

VII. Conclusion

Although a lot of ground has been covered in the world of artificial intelligence since Turing wrote his paper in 1950, it seems like the imitation game that he described is still an apt test for computer intelligence. Apt enough, at least, to defy modern researchers. The test is still contentious, and the objections that Turing brings up in his paper are remarkably prescient. Even though we have a much greater body of knowledge to draw from, the same few objections still generate a heated discussion. Although there are some flaws in Turing's predictions, that he could have anticipated the debate so completely is a testament to his insight.

References

[1] Turing, A.M. 1950. Computing machinery and intelligence. Mind, 59:433-460.

[2] Kurzweil, R. 1982. The Age of Intelligent Machines: A Kind of Turing Test.

[3] McCulloch, W. S. and Pitts, W. H. 1943. A logical calculus of the ideas immanent in nervous activity. Bulletin of Mathematical Biophysics, 5:115-133.

[4] Chomsky, N. 1957. Syntactic Structures, Mouton, The Hague.

[5] Friedberg, R. M. 1958. A learning machine: Part I. IBM Journal, 2(1):2-13.

[6] Minsky, M and Papert, S. (1968). Perceptrons: an Introduction to Computational Geometry. MIT Press.

[7] Pfeiffer, J. 1962. The Thinking Machine New York: Lippincott. 174.

[8] Dreyfus, H. L. 1972. What Computers Can't Do. MIT Press.

[9] Searle, J. R. 1980. Minds, brains, and programs. Behavioral and Brain Sciences, 1:417-24. Also in J. Haugeland (ed) (1997) Mind Design II. MIT Press: Cambridge, MA.

[10] Robson, J. M. 1984. N by N checkers is Exptime complete. SIAM Journal on Computing, 13(2):252-267.

[11] Fraenkel, A.S. and Lichtenstein, D. 1981. Computing a perfect strategy for n*n chess requires time exponential in n, Proc. 8th Int. Coll. Automata, Languages, and Programming, Springer LNCS 115:278-293 and J. Comb. Th. A 31:199-214.

[12] Brown, E., Kass, R., and P. Mitra. 2004. Multiple neural spike train data analysis: state-of-the-art and future challenges. Nature Neuroscience. 7(5):456-461.

[13] Gelman, Rochel, Gallistel, C. R. 2004. Language and the Origin of Numerical Concepts. Science 306: 441-443.

[14] Lehrer, J. News Hour with Jim Lehrer. May 12, 1997.

[15] Russell S, Norvig P. 1995. Artificial Intelligence: A Modern Approach, Prentice Hall Series in Artificial Intelligence. Englewood Cliffs, New Jersey. p948.