The Future of AI:

Can Machines Think?

Odysseas Diamantopoulos-Pantaleon

Research Paper

PH 4141 Philosophy of Mind

Dr. Elly Vintiadis

Deree – The American College of Greece

Spring 2020

Words: 3264

Introduction

Rapid advances in the field of Artificial Intelligence have challenged scientists and philosophers to redefine their views about the capabilities of machines. Many of them strongly believe that machines can in fact achieve way more things than what we previously thought. This has of course impacted the debate around the mind-body problem and has made some of the theories that surround it more attractive, like Functionalism. Functionalism expresses the belief that what makes thoughts, desires, anxiety and any other type of mental state, does not depend on its internal constitution, but on its functional role or better, on the role it performs in a certain cognitive system. For example, a functionalist would believe that pain is a state that tends to be caused when a body injury happens, which produces the belief that something is wrong with the body and that you should exit that state, resulting in moaning. In this example we see a basic principle of functionalist theories. More specifically, functionalists change the question from "what is it?" to "what does it do?" and introduce three kind of causal relations that are very common in computer science, input followed by processing that results into an output. In other words, input gathered through sensory devices is processed inside the system and results in a certain output. [1]Using our previous example, we can say that functionalism describes the mental state of pain in humans as input given by sensory devices, major or minor body injury sensed by the nerves in the body, that is processed by the organism and which results in a certain output, the urge to stop what you are doing that causes that pain or to yell and cry. Functionalism quickly grew as a theory and was embraced by many, especially scientists, because of the theory's ability to be compatible with both materialism and dualism, the two major theories fighting over the mind-body problem, and because of the similarities it had with the way modern science functions. Hillary Putnam in his papers in 1960 and 1967 first described a branch of functionalism, Machine State Functionalism, an endorsement of computational theories of the mind. According to Putnam, any organism that has a mind can be regarded as a Turing machine whose actions are defined by a certain set of instructions of the usual structure that is used in

[1] Janet Levin, "Functionalism", The Stanford Encyclopedia of Philosophy (Fall 2018 Edition), Edward N. Zalta (ed.), URL = <https://plato.stanford.edu/archives/fall2018/entries/functionalism/>, paragraph 1

functionalist theories, input that leads to processing that results in an output.[2] This theory was one of the first that implied that there is a possibility that a machine can produce thought, something that has now become an extremely popular idea which has inspired and motivated scientists to achieve various advances in many different scientific fields like cognitive science and artificial intelligence. However, is it possible to assume that computers can achieve thought even though they do not seem to be able to perform advanced cognitive processes, like memory and learning? The aim of this paper will be to explore the various arguments in favor of each side and through that try to establish my own view on the dilemma of "Can Machines Think"?

Janet Levin, "Functionalism", The Stanford Encyclopedia of Philosophy (Fall 2018 Edition), Edward N. Zalta (ed.), URL = <https://plato.stanford.edu/archives/fall2018/entries/functionalism/>, paragraph 3.1

Looking into Computers

Alan Mathison Turing, a brilliant scientist, is perhaps one of the most significant influencers of the computer – thought connection. Through his groundbreaking work in the fields of mathematics and computer science he managed to completely alter the general opinion regarding computers and their capabilities, inventing as many say the "Computer Age". In mathematics algorithms and computations are extremely important. An algorithm is basically a list of mechanical instructions that if followed step by step solve a problem. Turing in 1936 published a landmark paper "On Computable Numbers, With an Application to the Entscheidungs problem" which offered an extremely influential analysis on computations and introduced the so-called Turing machines. Turing machines are idealized computing devices that have unlimited storage space and time at their disposal. These devices manipulate symbols just like humans use pencil and paper to perform arithmetical computations.[3]The memory locations of this machine form a linear structure from which the central processor of the machine can access only one memory location at a given time. This processor can perform only four actions. It can write or erase a symbol in a memory location and access the next or the previous memory location in the linear array. The behavior of the machine is dependent on its machine table. The machine table dictates a finite set of routine instructions that control the system's reaction to input and that change its machine states. More specifically, the machine table is a collection of internal states that analytically explain to the machine how to act in each given case. Turing believed that humans have limits both on our perceptual and our cognitive mechanisms and cited that every possible humanly executable algorithm can be replicated by using a Turing machine, leaving space for dreaming a fully computational mind.

Tim Crane in his book *The Mechanical Mind* analyzed this idea of a computational mind and explained how Turing along with some other great scientists led us to the invention of general purpose computers that we use in our everyday life and which are considered by some capable of thinking. Crane openly questions the ability of machines to think. As he explains, most processes that happen in the brain are not as simple as the examples we gave before. In order for the Turing machine to be able to perform more complex actions it needs a more complex machine table. To be able to achieve that it is necessary to expand the machine table by adding more internal states, more tape, meaning more

[3] Liesbeth De Mol, "Turing Machines", The Stanford Encyclopedia of Philosophy (Winter 2019 Edition), Edward N. Zalta (ed.), URL = <https://plato.stanford.edu/archives/win2019/entries/turing-machine/>.

memory, and a more complicated alphabet. Using the binary alphabet, it is now possible to code the machine table itself as numbers in the notation and through that we can have a Turing machine taking the tape of another Turing machine as input. That means that we can have a Turing machine mimic the behavior of another Turing machine resulting in what we call, the universal Turing machine.
This is extremely exciting because it means that for every different operation that we might want to do, we can have a single machine that for every algorithm that we need to solve it just changes its behavior to match the Turing machine that corresponds to that operation. It is generally accepted that this machine is the ancestor of digital computers and that is why Turing is often credited as the inventor of the Computer Age. Another important note is that modern computers use automated algorithms where the input, the machine table and the output are built into their physical structure. That way there is no need for a human agent to intervene when for example, the tape needs to be fixed or readjusted, but at the same time they can be programmed just like the Universal Turing machine to mimic the operations of other machines.

We have now extensively analyzed how computers came to be and how they work. However, we have not yet explained how a computational system like a computer can execute common processes of the mind. After all, minds can achieve complicated thinking and do not limit themselves in simple algorithms. Some predecessors of Turing had already poured a lot of thought in the question of how we can represent complex statements in machine language. Gottfried Wilhelm Leibniz first proposed the idea of the universal character. The ‘characteristica universalis’ is supposed to be a mathematically precise and clear language in which human ideas and thoughts could be translated. Leibniz went ahead to design binary notation, the alphabet used to design more complex Turing machines, but, unfortunately, never managed to achieve the complexity behind what he named universal character. George Boole, however, in his book called ‘The Laws of Thought’, published in 1854, came even closer to their aspiration. Boole designed a new algebra in order to explain relations between statements, but it was not like the ordinary algebra. Boolean algebra sought to express elementary logical relations between statements using the simple words ‘or’, ‘and’ etc., and implement them in the binary notation. Boole believed that by building up these patterns of reasoning we can eventually uncover the fundamental laws of operations that the mind uses to reason. In other words, he wanted to codify human thought and uncover the secret mechanisms behind reasoning. His efforts resulted in an even more advanced alphabet, that along with the

binary notations created the modern machine language.[4]

[4] An extensive presentation of Turing Machines and their relationship with modern computers can be found in: Tim Crane, *The Mechanical Mind* (New York: Routledge, 2003), 83-133

Classical Computational Theory of the Mind

The revolutionary ideas of Turing, lead Warren McCulloch and Walter Pitts to first propose that a Turing machine can perhaps be a very good model for the human mind. This led to a family of views called Classic Computational Theory of Mind, also known as CCTM. The CCTM claims that the mind should not be referred as a computer, because a computer system is programmable.
It also holds that the mind does not just resemble a computing system, but it literally is one. The human mind is made up of flesh and blood whereas artificial computing systems are made of silicon chips and function with electricity. According to the CCTM this difference is just disguising the similarity these two have, which we can understand by looking at the Turing computational model. Through this we can reach an abstract computational model that both systems have in common and CCTM theorists use it to make the point that we can have a precise, true description and mapping of core mental processes.[5] This idea is rather enthralling since it opens the door to many possibilities, like extremely advanced Artificial Intelligence machines.

However, it is difficult to accept that the mind is purely computational. For example, it would be controversial to think that there is a computational theory of pain. These theories usually argue that the mental states themselves are not computational, but their relation is. In other words, there are algorithmic rules that govern these cognitive processes. That is an argument which is extremely difficult to prove and in order to be proven, we need to advance even further from the Boolean algebra and come closer to the creation of the 'characteristica universalis' that Leibniz introduced. Therefore, I cannot yet abide with the thesis that machines can think based entirely on the CCTM argument, since there is not enough scientific evidence to prove that our brain is purely computational or that mental processes are connected by some sort of an algorithm.

[5] Michael Rescorla, "The Computational Theory of Mind", The Stanford Encyclopedia of Philosophy (Spring 2020 Edition), Edward N. Zalta (ed.), URL = <https://plato.stanford.edu/archives/spr2020/entries/computational-mind/>.

Chinese Room Experiment and Origins

Turing's opinion on the subject was strongly expressed in his paper "Intelligent Machinery" on 1948. He makes use of what he calls a "Paper Machine", a piece of paper where there are written instructions for playing chess. More specifically, there are simple algorithmic steps written in a non-machine language, for example English, and that will be executed by a human being. As Turing explains, the human that executes all these instructions does not need to know of chess existence. He goes as far as stating that the agent does not even need to know that he is playing chess! There will be just input that the operator will have to manipulate to produce a certain output and these do not need to mean something to him. Furthermore, if the man is closed in a room with this paper and is playing a game of chess with people outside this room that do not know anything about the whole scenario, he will seem to them as an expert chess player. We can also include in the scenario the possibility that the operator has learned the paper program by heart and manipulates all the symbols inside his head. The question that all this raise is, does the man know how to play chess? Does someone's conscious states matter for knowing to play a game? Can we assume that if we implemented the same program in a digital computer, then the computer would play chess, or just simulate it? Turing strongly believed that in the future computers will start to exhibit intelligent behavior and for that reason he devised the Turing Test, a test used to determine if a computational system is intelligent.[6]

John Searle, however, did not agree with Turing and the Intelligent Machines that he spoke of. Searle believed that even if a machine can execute the same processes as a human being, it still wouldn't constitute of thinking because it lacks the understanding of what it is doing. Based on this Searle created The Chinese Room Experiment. Similar to the "Paper Machine", the Chinese Room Experiment exhibits an English native speaker who doesn't know Chinese, sitting alone in a room full of boxes of Chinese symbols and a book which helps him manipulate all these symbols. People from outside that room feed him with questions in Chinese and the goal of that man is to answer these questions based on the instructions he has. The boxes with the Chinese symbols resemble the database from which a program draws information, the book that the man uses to manipulate these symbols is the program, the questions that are given to the man are the input and the answers that the man gives

[6]David Cole, "The Chinese Room Argument", The Stanford Encyclopedia of Philosophy (Spring 2020 Edition), Edward N. Zalta (ed.), URL = <https://plato.stanford.edu/archives/spr2020/entries/chinese-room/>, paragraph 2.2

based on what he is instructed to do are the output. According to Searle, this man will be able to answer all these questions in Chinese and seem to know Chinese, without understanding a thing. Similarly, a machine will never be able to understand Chinese simply because there is no meaning behind manipulating symbols[7].

A very popular answer to the issues posed by Searle, was introduced by Block, Dennet and others, using the name of "The Systems Reply". The Systems Reply main point is that Searle makes a category mistake regarding the human inside the room. The operator inside the room, they say, shouldn't be identified as the whole system, but as a part of it, for example a CPU. As they explain, a CPU on its own is of no importance, but if you take the system as a whole, which includes the CPU, the database, the input etc., then you can generate understanding. So, if the system as a whole has the capacity to understand Chinese, then it should understand it. Margaret Boden goes even further and supports that we can not support that a brain understands English, because understanding is something exhibited by the whole organism, a person, and that the brain is no more than the basis of intelligence.[8]

Another reply to Searle which is interesting, is the "Robot Reply". This view concedes that the man in the room doesn't show understanding of Chinese. However, it states that in order to know the meaning of a Chinese word, you would have to experience it in the real world. Furthermore, it is stated that if a robot was given the necessary input devices with which to interact with the material world, the robot would be just like a newborn baby which starts learning by seeing and doing stuff in the real world. In other words, the supporters of this view would agree that symbols without any connection to the real world are insufficient for semantics but were they to relate to real life experiences they would probably yield understanding. Hilary Putnam's example of a "Brain in a Vat" contributed to this view and made clearer that meaning must be connected with experiences.[9]

I believe that agreeing with Searle on the Chinese Room Experiment will have a major consequence. If we agree that the man or the program that is answering the questions based on instructions has no understanding, then we have to concede that all people around the world speaking a certain language also have no understanding of that language. They are, just like a computer program, simply manipulating symbols that they have learnt by heart since birth. Therefore, I agree that syntax by

---

[7] David Cole, "The Chinese Room Argument", The Stanford Encyclopedia of Philosophy (Spring 2020 Edition), Edward N. Zalta (ed.), URL = <https://plato.stanford.edu/archives/spr2020/entries/chinese-room/>, paragraph 3
[8] David Cole, "The Chinese Room Argument", The Stanford Encyclopedia of Philosophy (Spring 2020 Edition), Edward N. Zalta (ed.), URL = <https://plato.stanford.edu/archives/spr2020/entries/chinese-room/>, paragraph 4.1
[9] David Cole,"The Chinese Room Argument", The Stanford Encyclopedia of Philosophy (Spring 2020 Edition), Edward N. Zalta (ed.), URL = <https://plato.stanford.edu/archives/spr2020/entries/chinese-room/>, paragraph 4.2

itself has no semantic, but if we experience the thing that we have chosen to represent with a certain symbol, this symbol can gain meaning. In more advanced languages we use a combination of symbols to represent, for example, a lion. Before seeing a lion, a human just knows the definition of what it is, there is no meaning behind the use of the word. He can use it to express hunger," I am hungry like a lion", because he has learned that this symbol can be manipulated this way to show that you are hungry. However, without experiencing it there will be no meaning behind its use. That is why I agree with the "Robot Reply" that there can be no meaning without experience.

I also agree with the argument expressed by the Systems Reply. I also believe that Searle has committed a category mistake, since indeed if we search for things in such a low level as the brain or the CPU of a computer, we will never find any understanding of the actions performed, since the chemical or electrical activities that happen inside these parts are just playing a role in the whole system. So, if you want to look for comprehension, you have to view the whole picture.

Conclusion

All in all, we can find many similarities between a computer system and a human being, which is natural since we are the ones that designed them in the first place. It is not easy to realize the extent of our accomplishment. We question whether both have similar structures or operate the same and we question the ability of these systems to reach the human level. The thing is that till this day we know very little about our own system and the insecurity caused by this lack of information makes us afraid of accepting new trivial concepts and realizing our feats. Every day we assume that the human next to us is the same as us, that he has feelings, dreams, and he is not some sort of a philosophical zombie, mainly because he looks like us. But if you think about it, intelligent non-human systems can be the same. We just instantly reject the possibility because they don't look like us or because they were created by us. I believe that in time, with technology advancing, we will be able to both learn more about ourselves and improve our machine creations, which will result in realizing that Machines can indeed produce thought.

Bibliography

Cole, David, "The Chinese Room Argument", The Stanford Encyclopedia of Philosophy (Spring 2020 Edition), Edward N. Zalta (ed.), URL = <https://plato.stanford.edu/archives/spr2020/entries/chinese-room/>.

Crane, Tim. *The Mechanical Mind*. New York: Routledge, 2003.

De Mol, Liesbeth, "Turing Machines", The Stanford Encyclopedia of Philosophy (Winter 2019 Edition), Edward N. Zalta (ed.), URL = <https://plato.stanford.edu/archives/win2019/entries/turing-machine/>.

Levin, Janet, "Functionalism", The Stanford Encyclopedia of Philosophy (Fall 2018 Edition), Edward N. Zalta (ed.), URL = <https://plato.stanford.edu/archives/fall2018/entries/functionalism/>.

Rescorla, Michael, "The Computational Theory of Mind", The Stanford Encyclopedia of Philosophy (Spring 2020 Edition), Edward N. Zalta (ed.), URL = <https://plato.stanford.edu/archives/spr2020/entries/computational-mind/>.