Get a Sneak Peek of the New *Non-Computable You* Audiobook

https://mindmatters.ai/podcast/ep207

Austin Egbert:

Greetings and welcome to Mind Matters News. Non-Computable You by our host Robert J. Marks is now available in audiobook form on Audible at amazon.com. Here's an excerpt from chapter one as read by Larry Nobles. Enjoy.

Larry Nobles:

Part One: Brick Walls AI Will Never Go Through; Chapter 1, The Non-Computable Human. Our first successful humanoid robot, the first robot that is clearly on the road to a human-like imitation mind won't happen until we know how to imitate human emotions and how to integrate them completely into artificial thought. Of course, such robots will feel nothing. We have no way to make a computer or any machine feel and we probably never will. David Gelernter, Yale University. If you memorized all of Wikipedia, would you be more intelligent? It depends on how you define intelligence. Consider John Jay Osborn Jr's 1971 novel, The Paper Chase. In this semi-autobiographical story about Harvard Law School, students are deathly afraid of Professor Kingsfield's course on contract law. Kingsfield's classroom presence elicits both awe and fear. He is the all knowing professor with the power to make or break every student.

Larry Nobles:

He is demanding, uncompromising and scary smart. In the iconic film adaptation, Kingsfield walks into the room on the first day of class, puts his notes down, turns toward his students and looms threateningly. "You come in here with a skull full of mush." He says, "You leave thinking like a lawyer." Kingsfield is promising to teach his students to be intelligent like he is. One of the law students in Kingsfield's class, Kevin Brooks, is gifted with a photographic memory. He can read complicated case law and after one reading, recite it word or word. Quite an asset, right? Not necessarily. Brooks has a host of facts at his fingertips, but he doesn't have the analytic skills to use those facts in any meaningful way. Kevin Brooks' wife is supportive of his efforts at school and so are his classmates, but this doesn't help. A tutor doesn't help. Although he tries, Brooks simply doesn't have what it takes to put his phenomenal memorization skills to effective use in Kingsfield's class.

Larry Nobles:

Brooks holds, in his hands, a million facts, that because of his lack of understanding are essentially useless. He flounders in his academic endeavor. He becomes despondent, eventually he attempts suicide. This sad tale highlights the difference between knowledge and intelligence. Kevin Brooks' brain stored every jot and tittle of every legal case assigned by Kingsfield, but he couldn't apply the information meaningfully. Memorization of a lot of knowledge didn't make Brooks intelligent in the way that Kingsfield and the successful students were intelligent. British journalist Miles Kington captured this distinction when he said, "Knowing a tomato is a fruit is knowledge. Intelligence is knowing not to include it in the fruit salad." Which brings us to the point when discussing artificial intelligence, it's crucial to define intelligence. Like Kevin Brooks, computers can store oceans of facts and correlations, but intelligence requires more than facts.

Larry Nobles:

True intelligence requires a host of analytic skills. It requires understanding, the ability to recognize humor, subtleties of meaning and symbolism, and the ability to recognize and disentangle ambiguities. It requires creativity. Artificial intelligence has done many remarkable things, some of which we'll discuss in this book. AI has largely replaced travel agents, toll booth attendants and map makers. But will AI ever replace attorneys, physicians, military strategists, and design engineers, among others? The answer is no. And the reason is that as impressive as artificial intelligence is, and make no mistake, it is fantastically impressive. It doesn't hold a candle to human intelligence. It doesn't hold a candle to you and it never will. How do we know? The answer can be stated in a single four syllable word that needs unpacking before we can contemplate the non-computable you. That word is algorithm. If not expressible as an algorithm, a task is not computable.

Larry Nobles:

Algorithms and the computable. An algorithm is a step by step set of instructions to accomplish a task. A recipe for German chocolate cake is an algorithm. The list of ingredients acts as the input for the algorithm. Mixing the ingredients and following the baking and icing instructions will result in a cake. Likewise, when I give instructions to get to my house, I'm offering an algorithm to follow. You're told how far to go and which direction you're to turn on what street. When Google Maps returns a route to your destination, it's giving you an algorithm to follow. Humans are used to thinking in terms of algorithms. We make grocery lists. We go through the morning procedure of showering, hair combining, teeth brushing, and we keep a schedule of what to do today. Routine is algorithmic. Engineers algorithmically apply Newton's laws of physics when designing highway bridges and airplanes. Construction plans captured on blueprints are part of an algorithm for building.

Larry Nobles:

Likewise, chemical reactions follow algorithms discovered by chemists and all mathematical proofs are algorithmic. They follow step by step procedures built on the foundations of logic and axiomatic presuppositions. Algorithms need not be fixed. They can contain stochastic elements such as descriptions of random events and population genetics and weather forecasting. The board game Monopoly, for example, follows a fixed set of rules. But the game unfolds through random dice throws and player decisions. Here's the key: computers only do what they're programmed by humans to do, and those programs are all algorithms, step by step procedures contributing to the performance of some task. But algorithms are limited in what they can do. That means computers, limited to following algorithmic software, are limited in what they can do. This limitation is captured by the very word computer. In the world of programmers, algorithmic and computable are often used interchangeably. And since algorithmic and computable are synonyms, so are non-computable and non-algorithmic. Basically for computers, for artificial intelligence, there's no other game in town. All computer programs are algorithms. Anything non-algorithmic is non-computable and beyond the reach of ai, but it's not beyond you.

Larry Nobles:

Non-computable you. Humans can behave and respond non-algorithmically. You do so every day. For example, you perform a non-algorithmic task when you bite into a lemon. The lemon juice squirts into your tongue and you wince at the sour flavor. Now consider this. Can you fully convey your experience to a man who was born with no sense of taste or smell? No, you cannot. The goal is not a description of the lemon biting experience, but its duplication. The lemon's chemicals and the mechanics of the bite can be described to the man, but the true experience of the lemon taste and aroma can't be conveyed to someone without the necessary senses. If biting into a lemon can't be explained to a man without all his functioning senses, it certainly can't be duplicated in an experiential way by AI using computer software. Like the man born with no sense of taste or smell, machines do not possess qualia, experientially sensory perceptions, such as pain, taste, and smell.

Larry Nobles:

Qualia are a simple example of the many human attributes that escape algorithmic description. If you can't formulate an algorithm explaining your lemon biting experience, you can't write software to duplicate the experience in the computer. Or consider another example, I broke my wrist a few years ago and the physician in the emergency room had to set the broken bones. I'd heard beforehand that bone setting really hurts, but hearing about pain and experiencing pain are quite different. To set my broken wrist, the emergency physician grabbed my hand and arm, pulled and there was an audible crunching sound as the bones around my wrist realigned. It hurt, a lot. I envied my preteen grandson who had been anesthetized when his broken leg was set. He slept through his pain. Is it possible to write a computer program to duplicate, not describe, but duplicate my pain?

Larry Nobles:

No, qualia are not computable. They're non-algorithmic. By definition, and in practice, computers function using algorithms. Logically speaking then, the existence of the non-algorithmic suggest there are limits to what computers and therefore AI can do. The software of the gaps. There are other human characteristics that cannot be duplicated by AI. Emotions such as love, compassion, empathy, sadness, and happiness cannot be duplicated nor can traits such as understanding, creativity, sentience and consciousness. Or can they? Extreme AI champions argue that qualia and indeed all human traits will someday be duplicated by AI. They insist that while we're not there yet, the current development of AI indicates we will be there soon. These proponents are appealing to the software of the gaps, a secular cousin of the God of the gaps. Machine intelligence, they claim, will someday have the proper code to duplicate all human attributes. Impersonate perhaps, but experience, no.

Larry Nobles:

Mimicry versus experience. AI will never be creative or have understanding. Machines may mimic certain other human traits but will never duplicate them. AI can be programmed only to assimilate love, compassion, and understanding. The simulation of AI love is wonderfully depicted by a human appearing robot boy, brilliantly acted by a young Haley Joel Osment in Steven Spielberg's 2001 movie A.I. Artificial Intelligence. Before activation, the robot boy, played by Osment, is emotionless. But when his love simulation software is turned on, the boy's immediate attraction to his adoptive mother is convincing. Thanks to Osment's marvelous acting skill, the robot boy is attentive, submissive and full of snuggle love. But mimicking love is not love. Computers do not experience emotion. I can write a simple program to have a computer enthusiastically say I love you and draw a smiley face, but the computer feels nothing. AI that mimics should not be confused with the real thing.

Larry Nobles:

Emergent consciousness. Moreover, tomorrow's AI, no matter what is achieved, will be from computer code written by human programmers. Programmers tap into their creativity when writing code. All computer code is the result of human creativity. The written code itself can never be a source of creativity itself. The computer will perform it as it is instructed by the programmer, but some hold that as code becomes more and more complex, human like emergent attributes such as consciousness will appear. Emergent means that an entity develops properties that its parts don't have on their own. A sum greater than the parts can account for. This is something called strong AI. Those who believe in the coming of strong AI argue that non-algorithmic consciousness will be an emergent property as AI complexity ever increases. In other words, consciousness will just happen as a sort of natural outgrowth of the codes increasing complexity. Such unfounded optimism is akin to that of a naive young boy standing in front of a large pile of horse manure.

Larry Nobles:

He becomes excited and begins digging into the pile, flinging handfuls of manure over his shoulders. "With all this horse poop," he says, "there must be a pony in here somewhere." Strong AI proponents similarly claim, in essence, with all this computational complexity, there must be some consciousness here somewhere. There is, the consciousness residing in the mind of the human programmer. But consciousness does not reside in the code itself and it doesn't exchange from the code any more than a pony will emerge from a pile of manure. Like the boy flinging horse poop over his shoulder, strong AI proponents, no matter how insistently optimistic, will be disappointed. There is no pony in the manure. There is no consciousness in the code. Uploading a brain. Are there any similarities between human brains and computers? Sure. Humans can perform algorithmic operations. We can add a column of numbers like a computer, though not as fast.

Larry Nobles:

We learn, recognize, and remember faces and so can AI. AI, unlike me, never forgets a face. Because of these types of similarities, some believe that once technology has further advanced and once enough memory storage is available, uploading the brain should work. Whole brain emulation also called mind upload or brain upload is the idea that at some point we should be able to scan a human brain and copy it to a computer. The deal breaker for whole brain emulation is that much of you is non-computable. This fact nixes any ability to upload your mind into a computer. For the same reason that a computer can't be programmed to experience qualia, our ability to experience qualia cannot be uploaded to a computer. Only our algorithmic part can be uploaded and an uploaded entity that is totally algorithmic, lacking the non-computable, would not be a person, so don't count on digital immortality. There are other more credible roads to eternal life.

Larry Nobles:

Understanding and Searle's Chinese room. An IBM computer program dubbed Watson, famously took on two world champions on the quiz show, Jeopardy. Watson was named after an IBM executive and not after the sidekick of Sherlock Holmes. Watson gave the correct responses to many of the queries asked on the show. The computer program had access to all of Wikipedia and then some. But does IBM's Watson understand what it is doing when sifting through tones of data to find the right answer? Does Watson understand either the queries it receives or the answers it gives? Philosopher John Searle says no. Searle illustrates this convincingly with a first person parable about being isolated in a large room. Also in the room are many file cabinets containing Chinese prose. The Chinese room accepts questions in Chinese slipped through a slot in the door. Searle, isolated in the room with his file cabinets, doesn't understand Chinese. But armed with the slip of paper from outside, Searle begins searching through the many stuffed file cabinets.

Larry Nobles:

His goal is to match the Chinese question written on the paper to an entry stored somewhere in the file cabinets. After some exploring, he finds the match on a filed index card. Also, on the card written in Chinese is the response to the submitted query. Searle copies the response on the back of the slip of paper, returns the card to the file cabinet and slips the paper with the response out the slot in the door. From the outside, it looks like Searle understands Chinese. After all, the question was submitted in writing using Chinese and the response is written in Chinese. But Searle doesn't know Chinese. He can either read nor understand Chinese. Likewise, a computer doesn't understand what it is doing. A computer operates, as in the Chinese room parable, using algorithms. Computers are queried and supply answers, but they have no understanding of what they're doing. IBM's Watson is simply a humongous Chinese room using a Wikipedia like database for its file cabinets. Watson gives Jeopardy answers but has no understanding of what the questions and answers mean. We'll return to Watson shortly.

Larry Nobles:

Now, however, let's look at other examples of behaviors that give the impression of intelligence while the agent in fact lacks understanding.

Austin Egbert:

You've been listening to an excerpt from chapter one of Non-Computable You by Robert J. Marks. You can find the rest in print, Kindle, and audio formats on amazon.com. And be sure to leave us an enthusiastic five star review so we know you enjoyed it. Thanks for listening and until next time, be of good cheer.