Artificial Intelligence

The Washington Post reports that artificial intelligence (AI) chatbots "hallucinate." That is, they are so driven to respond and to value response over everything else that they lie *and* experience a form of self-denial in which they come to accept those lies as truth. Examples of this are appearing daily in the news.

An attorney in Manhattan used a chatbot to write his legal complaint and specifically asked the chatbot to verify the sources. The chatbot responded sincerely that all the sources were verified, but the attorney unfortunately did not know any better. At trial, the judge determined that the precedents cited in the brief were entirely fictitious.

The U.S. Air Force conducted a simulation with an AI piloted drone. The drone had instructions to destroy a target, but when the drone determined that the human operator was impeding the mission, the drone destroyed the communications link to the operator and then "killed" the operator.

These hallucinations are not surprising. A scan of the research literature finds that this behavior has been known to the research community for at least five years. OpenAI CEO Sam Altman recently testified to Congress that AI poses an existential threat to the world. So, I think I am safe to ask, what idiot developed this software? How could someone, or multiple people, devote so many resources and aggressively distribute a technology that is not merely flawed, but dangerous?

Unfortunately, the answer from me is a little bit of, "I told you so." The flawed introduction of AI can be traced to excessive trust in technical engineers and the simultaneous weakening of the education of those same engineers. Technical entrepreneurs have been given an unquestioning deference and loyalty that excuses them from common performance measures. This is in complete contrast to most other science professions. Doctors and the medical profession, for example, are regularly challenged in their advice and guidance. Climate science is regularly disputed.

There was a time in my father's generation when doctors, professors and other learned people could effectively get away with anything so long as they dressed like a scholar and smoked a pipe. This was a bad extreme. A lot of bad decisions and bad personal behavior were excused. Robert Moses paved over city parks and destroyed neighborhoods in New York City with his highway plans but was toasted at all the fine cocktail parties.

IT entrepreneurs are now the recipients of this deference. Their fabulous wealth and enormous armies of employees give us pause, but they developed this wealth on our labor. They didn't rigorously test their products and then back them up with a guarantee. They distributed their buggy, incomplete products and then made the public pay for the privilege. Did you know that Microsoft once had a music DJ for their telephone help line? Thousands of people were on hold, often for hours

at a time (I was one of them), waiting to speak to technical support, so Microsoft created their own closed circuit broadcast station.

And IT entrepreneurs are notorious for selling incomplete or flawed products. In the go-go 1980s, there was a term for it, "vaporware." One product for the Macintosh computer called "Wingz" was aggressively marketed at trade shows with extravagant booths and promotional gifts, but the product itself did not exist.

How do IT entrepreneurs continue to get away with this? Well, for one, software is not a "product." It is a creative work protected under copyright law. Every single computer and application that you use is included in a license agreement that you approved and joined when you clicked "OK" at installation. It specifically says that you are licensing a creative work and that the creator accepts no liability for its performance. So, as you can imagine, there are no guardrails. There is no incentive for a software developer to test their work or establish any level of quality control.

The other problem we face is that today's technology engineers do not receive a balanced education. The great IT moguls themselves, such as Bill Gates, Steve Jobs and Mark Zuckerberg, take pride in having dropped out of their college educations. Many engineers do not receive any liberal arts education or topics outside of their technical field.

I have on numerous occasions reviewed publications or attended presentations in which it was obvious that the author had no knowledge of the industry or process, yet they intended to improve it. More than 10 years ago, I attended a presentation at a data science conference about a mathematical planning model for truck transportation. The audience included numerous industry professionals, and at the end of the presentation, they all said, "but this is not how our business operates." About six years ago, I attended a presentation on optimization of railway signaling, and the young researcher presented a solution that was entirely dependent on the train having a specific length in meters. I raised my hand at the end, "But what if the train length changes?" He asserted that the train length would not change, but I knew this to be incorrect, because as president of the local railway preservation society, I had driven our train on this very railway.

How can one write an all-encompassing AI software that is going to make decisions of life and death, or validate truth or lies, and not have any education on the issues of morality or culture? In the rush to make students more "job market ready," education programs have been stripped of all elements that are viewed as distractions to employment. But in a further economy move, many education programs that are not considered "sexy" have also been discontinued. My own field of transportation is an example. Many programs have been closed, including the graduate transportation program at my own university (the last one in Denmark). The argument has been made that the students can gain this applied business knowledge from their future employer.

The proposed solution to AI hallucinations is really going to make you laugh – or cry. The most recently reported solution is for the same question to be posed to TWO chatbots, and then have the two chatbots debate over which solution is correct.

Once again, I am playing devil's advocate, but if you have two identical pieces of software with the same data library, why would they deliver different answers? And if you are going to have them "debate" with each other, how do you prevent the chatbot with the wrong answer from convincing the chatbot with the correct answer to accept the wrong answer? What if both chatbots have the wrong answer at the start?

This reminds me of a childhood episode of "Dr. Who." The Doctor needed to find the safe exit, which was guarded by two robots. One exit led to safety, the other to certain death. One robot was programmed to always tell the truth, and the other robot always told a lie. Somehow, the Doctor posed a question to one robot on what the other robot would say, and that lead to the correct door. Is our future AI policy based on "Dr. Who"?

I certainly hope not. But more importantly, I do not subscribe to the belief that flaws in technology should be resolved with yet another layer of technology. This discussion of AI is entirely missing a root cause analysis. I believe there are two high-priority policy changes required. First, the days of software being excused from the classification as a product should be ended. Software is not a song on Spotify, a Harry Potter novel or those cat pictures you posted on Facebook. It is a real-world physical product that has life-and-death consequences, and it should be held to the same regulation and liability standards as a physical product.

Second, our engineers need a broader education beyond coding in Python and solving graph theory problems. They need to know the businesses and people for whom they will be making products. They need to know how these businesses function and deliver their services, and about their people and culture. They need to understand how one makes qualitative and moral decisions about truth and justice before they start writing AI software to evaluate medical diagnoses, argue court cases and fight wars.

Editor's note. The opinions expressed are that of the author and do not necessarily reflect those of *Analytics* magazine or the INFORMS organization.

Steven S. Harrod
(stehar@dtu.dk)
Steven S. Harrod is an associate professor at the Technical University of Denmark. He is a teacher and researcher in transport, especially railway transport. He is a member of INFORMS.