

Solving AI: We Need a New Language for Artificial Intelligence

Pedro Domingos

Department of Computer Science and Engineering
University of Washington

The goal of artificial intelligence (at least according to the field’s founders) is to create computers whose intelligence equals or surpasses humans’. Achieving this goal is the famous “AI problem.” To some, AI is the manifest destiny of computer science. To others, it’s a failure: clearly, the AI problem is nowhere near being solved. Why? For the most part, the answer is simple: no one is really trying to solve it. This may come as a surprise to people outside the field. What have all those AI researchers been doing all these years? The reality is that they have largely given up on the grand ambitions of AI and are instead working on increasingly specialized subproblems: not just machine learning or natural-language understanding, say, but issues within those areas, like classifying objects or parsing sentences.

I think that this “divide and conquer” approach won’t work. In AI, the best solution to a problem viewed in isolation often gets in the way of solving the larger problem. To make real progress, we need to work on “end to end” problems—self-contained tasks, like reading text and answering questions, that entail a number of subtasks. Until now, it hasn’t really been possible to do this, because the necessary computing power was not available. But within a decade or so, computers will surpass the computing power of the human brain. (While computers are extremely efficient at specific tasks, such as arithmetic, human brains are still ahead in terms of the number of operations they can perform per second. When this is applied to things that people are good at, like vision and language understanding, computers lose.)

Computing power is not the whole answer, though. Previous attempts to solve end-to-end AI problems have failed in one of two ways. Some oversimplified the problems to the point that the solutions did not transfer to the real world. Others ran into a wall of engineering complexity: too many things to put together, too many interactions between them, too many bugs.

To do better, we need a new mathematical language for artificial intelligence. Examples from other fields of science and technology demonstrate just how powerful this can be: mechanics, for example, benefited from calculus; alternating current from complex numbers; and digital circuits from Boolean logic. Today these things seem like second nature to their practitioners, but at the time they were far from obvious. The key is finding the right language in which to formulate and solve problems.

What should be the language of AI? At the least, we need a language that combines logic and probability. Logic can handle the complexity of the real world—large numbers of interacting objects, say, or multiple types of objects—but not its uncertainty. Probabilistic graphical models have emerged as a general language for dealing with uncertainty, but they can’t handle real-world complexity.

The last decade has seen real progress in this direction, but these are still early days. It’s unlikely that we’ll find the language of AI until we have more experience with end-to-end AI problems. But this is how we’re ultimately going to solve AI: through the interplay between addressing real problems and inventing a language that makes them simpler.