Go, Stop, Go, Stop

David Alan Grier, George Washington University

We’re dissecting the accomplishments of the past to more efficiently achieve the goals of the future.

It’s rarely a good idea to start a class with an announcement about some recent technological achievement that addresses an old and particularly sticky problem. If your students didn’t know the news before they walked into the room, they’re unlikely to appreciate its significance by the time they leave. At best, they’ll jot down a note about a bit of trivia that might appear on the final exam. At worst, they’ll assume you were trying to make them appreciate the accomplishments of an older generation. The world is a better place, they hear, because our elders had to trudge through the snow to get to school and labored long and hard through the night to replace an algorithm with complexity of only $n^{1.7}$.

The news that a computer program recently played and won a game of Go confirmed this pattern. My students stared blankly as I explained that Go is both simpler than chess and harder to automate. It’s played on a bigger board and has a more ambiguous resolution because it ends only when neither player sees any advantage in continuing. But these facts don’t impress a generation that was born when IBM’s Deep Blue beat world chess champion Garry Kasparov. To them, computers have always been able to play chess and play it well.

The basic algorithm that lies behind chess-playing programs transforms a demanding game into the task of searching through a very large set of potential moves and identifying the moves that best limit the opponent’s options. This algorithm has had limited success in other domains and has taught us little about the way in which humans play chess. Computers and humans “are playing an entirely different game,” observed Nathan Ensmenger, editor in chief of IEEE Annals of the History of Computing. As a result, computer chess has come “to be seen as increasingly distinct from human chess.”

In Science, it was argued that the new Go program doesn’t create as large a gap between the human and machine versions of the game as chess programs have. The Go program “won, not by virtue of overwhelming computational power,” it reported, “but by employing ‘machine learning’ tools that enable it to teach itself and to think more like humans do.”

These statements aren’t as straightforward as the periodical would have us believe. If the program reveals anything about how humans think, it’s that there’s a split between knowledge that can be analyzed or expressed in symbolic systems and knowledge that can only be captured as patterns. The Go program does a substantial amount of symbolic computing, and it also searches through potential moves, though it needn’t search as far. To simplify the search, it identifies likely moves through machine-learning algorithms that are ultimately rooted in statistical methods. The program determines the move that an opponent is likely to make, but it doesn’t distinguish whether such moves are motivated by optimal strategies, historic traditions, or the impulse of the moment.

Of course, it isn’t really important that a new Go-playing program thinks more like a human than an old chess program. The accomplishments of the past 40 years have not only changed the way we think about games, but also the way we approach intelligent machines. In the current age, artificial intelligence is less about building minds and more about taking apart boxes. We’re dissecting the accomplishments of the past to more efficiently achieve the goals of the future. We’ve always known that artificial intelligence plays a game that’s slightly different from that of natural intelligence, but our students will be the ones to tell us whether that actually matters.

DAVID ALAN GRIER is an associate professor of international science and technology policy at George Washington University. Contact him at grier@gwu.edu.