Why gaming AI won't help make AI work in the real world—but could
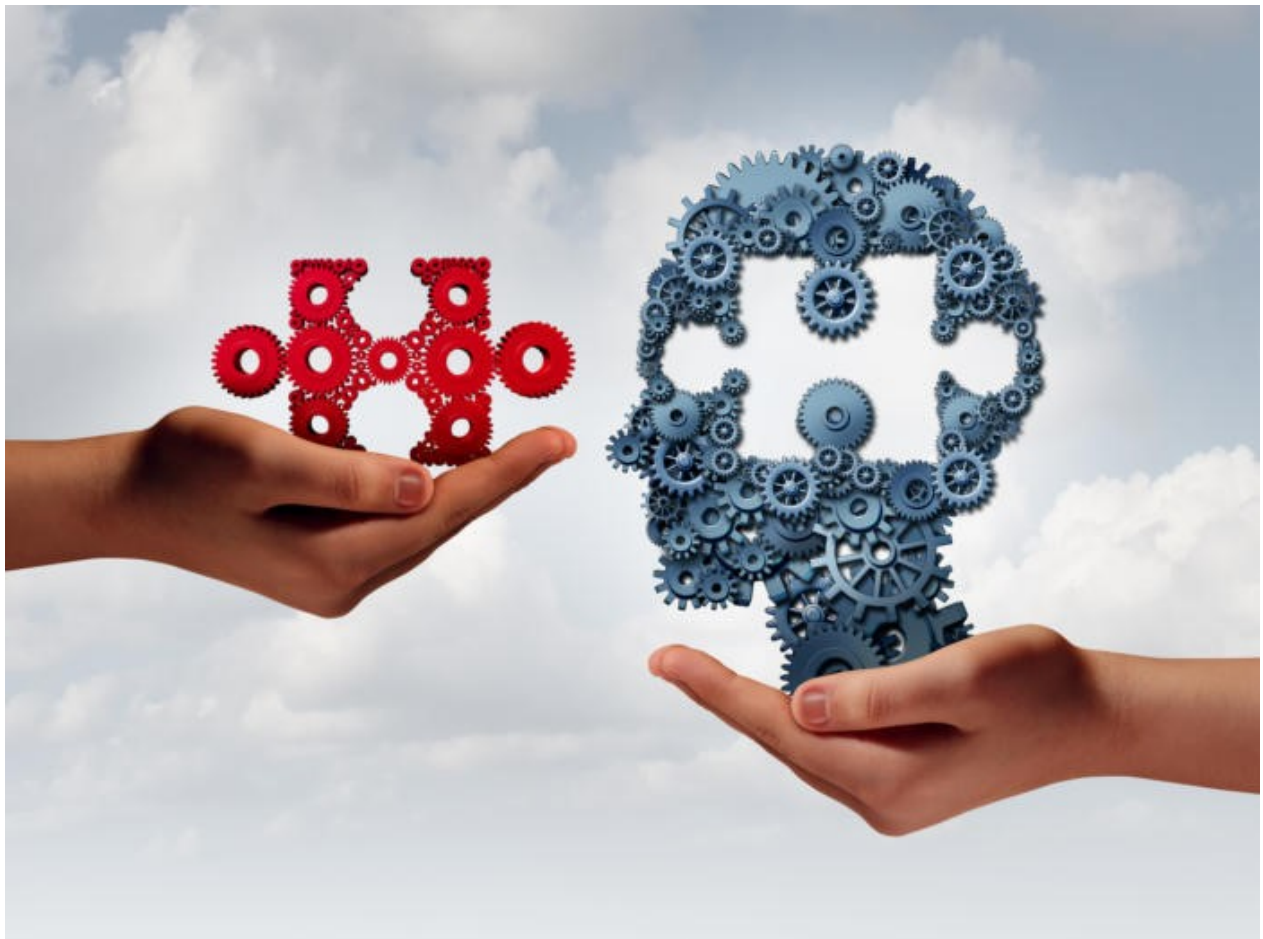
James Kobielus

The underlying game-theory principles could be applied to the real world, only if the test cases reflected the real world, not artificial fantasy environments

 By

Contributor, InfoWorld | Aug 30, 2018

Thinkstock

Multiplayer games are seen as a fruitful arena in which to simulate many real-world AI application scenarios—such as autonomous vehicles, swarming drones, and collaborative commerce—that may be too expensive, speculative, or risky to test fully in the real world. And they're an environment in which researchers can explore the frontiers of game theory, decision theory, and other branches of advanced mathematics and the computational sciences.

But when you dig deeper into AI in gaming, it becomes obvious that very little translates to the real world, so be very cautious of any business-oriented AI that comes out gaming contexts. But the underlying game-theory principles could be applied to the real world, only if the test cases

reflected the real world, not artificial fantasy environments.

OpenAI Five is a prime example of gaming-based AI's dubious promises

AI-centric gaming testbeds are all the rage. Fun and games are a great thing, but it remains to be seen whether AI researchers can apply the lessons learned in their design to problems in the real world. In particular, I'm thinking of OpenAI, the nonprofit research community that's pushing boundaries in artificial general intelligence.

I'm a bit concerned about the hoopla surrounding the ongoing OpenAI Five research project, which has been hyped for its supposed ability to "self-train" by using AI to play itself in multiple iterations of a multiplayer game called Defense of the Ancients (DotA) 2. We've seen these kinds of AI-enriched gaming initiatives for many years, but it's not clear whether any of them has produced any significant breakthroughs in new AI approaches that can spawn new types of applications beyond the narrowly constrained gaming domains for which they were developed.

I think Bill Gates and others are jumping the gun when they describe this sort of testbed as a milestone in ensuring "as many people benefit from AI as possible."Sure, there's entertainment value in these sorts of gaming-specific advances, and I assume that plenty of people will monetize these opportunities to the hilt. But that's not the same as pushing transferable uses that can solve real needs facing the human race.

After all, IBM Watson's *Jeopardy* coming-out was little more than a quiz-show gaming marketing stunt, and DeepMind's AlphaGo was merely the latest clichéd board-gaming face-off between an AI-driven bot and a human expert. Yes, those were impressive projects, but only if you think that the best minds of a generation are well-employed answering lame pop-trivia questions and pushing stupid little game pieces around a gridded board.

Personally, I'm not into gaming (though I'm a *Jeopardy* geek par excellence), so excuse me if I suppress a yawn. But here's what I gather about DotA 2 from the online coverage of the OpenAI Five project:

- In existence for more than a decade now, DotA 2 is a popular e-sport that some people practice constantly in hopes of nabbing a $40 million prize poolthat is said to be the largest such jackpot that can be won in any such online game right now.
- DotA 2 is a complex, continuous, real-world interactive strategy games that is played online.
- Players may be human beings or AI-driven neural-network agents.
- Each DotA player controls a character called a "hero."
- Players face off against each other in two teams of five.
- Players must collaborate with others to advance individual or team goals.
- Each player plans and executes strategy over long time horizons, striving to achieve predefined individual and/or team incentives.
- Players may take myriad actions across an exceedingly complex playing environment.
- Players must infer what enemies are doing and are likely to do without full information on the entire state of the game at any point in time.

Lest we chalk this up as purely fun and games, AI-style, it's clear that there's a more serious purpose driving OpenAI Five initiative. From what I can gather, one of OpenAI's key objectives is to generalize any lessons that may be learned from using AI to master DotA 2 so that they can be applied more generally to both gaming and nongaming application environments.

But perhaps I'm being too charitable, because the OpenAI team doesn't stray much beyond online gaming in its stated problem scope. In the broader AI landscape, the real endgame is the potential for *transfer learning*. This approach—also known as knowledge transfer, inductive transfer, and metalearning—refers to the possible reuse of some or all of the statistical knowledge from this AI application domain—in the form of training data, feature representations, neural-node layering, weights, training method, loss function, learning rate, and

other artifacts—in similar domains to address similar but not identical requirements.

Before I delve into the AI plumbing driving the OpenAI Five challenge, let's examine the overall solution domain to which these techniques have been applied, and ask how generally applicable they are to other problem spaces. Statistical learning can best be reused if the high-level problem definition has enough connection points to other human endeavors that may benefit from sophisticated AI.

At a high level, OpenAI Five is a learning challenge in which people and bots engage in complex, interactive, real-time, and team-oriented interactions that involve:

- **Real-time continuous interactions with a mix of strategic and inconsequential moves over long time-horizons.** DotA 2 moves like a blue streak over a considerable period of time. DotA 2 games run at 30 frames per second for an average of 45 minutes. AI-driven agents observe and respond to every fourth frame, with each agent often taking as many as 20,000 moves per game. By contrast, human players respond much more slowly and take many fewer moves with their respective heroes. Some, but not all, individual actions by a player/hero—human or agent—can affect the game strategically. Some strategies are short-lived while others play out over an entire game. By contrast, chess, Go, and many other traditional games involve far fewer moves, with almost every move in those games being strategic.
- **High-dimensional action and observation spaces.**DotA 2 is an exceptionally complex environment in which to plan, execute, and adapt strategy. DotA 2 players can each take dozens of actions and play on a large map containing ten heroes, dozens of buildings, dozens of nonplayer characters, and such game features such as runes and trees. In each tick of the game clock, there are 170,000 possible actions per player, which is much larger than chess's average of 35 possible actions and Go's average of 250 possible actions per move. Likewise, the observable state of a DotA game (the maximum amount of information a player is allowed to access at any point in time) is represented by 20,000 floating-point numbers. By contrast, chess and Go boards each have a far smaller and simpler observation space.
- **Partially observed states.**DotA 2 is shrouded in a digital simulacrum of what's often called the "fog of war." Dota 2 players can only see the units and buildings in the area around them, while the rest of the map is obscured. Frequently, players cannot see enemies in hiding, nor have enough information to calculate or anticipate adversaries' strategies with high confidence. Often, players must make inferences based on incomplete data and model what their opponents might be up to based on sketchy information. This contrasts with both chess and Go, which present all information on game state to both players at all times.

What, if any, value can be statistically "learned" from such a convoluted, highly specific set of gaming rules? What sort of AI "statistical knowledge" can transferred that's valid and useful in other gaming, application, and real-world contexts?

That's a huge fundamental issue with these sorts of AI gaming testbeds. Not only is one gaming environment often fundamentally different from others (for example, DotA 2 is most certainly not equivalent to a fantasy role-playing exercise such as Dungeons & Dragons), but gaming is quite different from other types of digital applications.

The differences between games of any sort and the real-world scenarios that they ostensibly mirror can be vast. For example, classic war games such as chess—and newbie creations such as DotA 2—are so far abstracted from actual armed human conflict that only a fool would try to use their algorithmic simulacra to control actual weapons, reconnaissance assets, and other military systems.

Once you grapple with that fundamental issue, you can't help but realize that the AI plumbing of a testbed is a secondary—albeit very interesting—matter. For the record, here's how OpenAI Five operates:

- In prior online versions, DotA 2's logic was implemented entirely in hundreds of thousands

of lines of server-based deterministic program code. The game's clock is driven by something called "tickrate," which is the frequency at which the DotA 2 server makes computations and sends them as update packets to the participants' computers or, in the case of bots, respective neural networks. In OpenAI Five, the bot players are individual statistically driven neural nets and the tickrates are 30 frames per second. It uses Kubernetes for internode orchestrated communications across 256 P100 GPUs and 128,000 [preemptible](#) CPU cores on Google Cloud Platform. From all agents/heroes per day, OpenAI Five's parallel-processing AI fabric collects the equivalent of 180 person-years of experience and processes around 37KB of observations. It collects observations at 7.5 observations per second of gameplay at around 60 observation batches per minute at a per-batch size of over a million observations. It processes around 150 to 170 actions per minute and has an average reaction time of 80ms.

- OpenAI Five uses AI-driven bots as "rollout workers," "evaluation workers," and "optimizer nodes." Rollout workers, which use around 500 CPUs running in parallel, execute a copy of the game as well as an agent that gathers experience. In each run of the game, evaluation workers, which use around 2,500 CPUs in parallel, assess each trained agent against reference agents, using such AI monitoring software such as [TensorBoard](#), [Sentry](#), and [Grafana](#). Optimizer nodes—which run on GPUs—drive training through synchronous gradient descent. This involves the optimizer nodes using GPUs to compute and globally average gradients using parallel processing and low-latency real-time cross-node data transfers. Worker and optimize nodes synchronize their experiences through Redis.

- AI-powered agents have access to the same information as human players of Dota 2, but the agents instantly see data that people have to check manually. Each agent embeds a unique, single-layer, 1024-unit [long short term memory](#) neural net that sees the current game state, computes actions as any of several "action heads," which correspond to a list of eight enumeration values and has semantic meaning, such as the number of ticks of the game clock to delay an action, the action to select, and the coordinates of the action. There is no explicit communication channel between the agents' individual neural networks.

- The game trains agents using a general-purpose reinforcement learning system called Rapid, which can be applied to any OpenAI [Gym](#) RL environment. Training uses a massively scaled-up version of a reinforcement-learning algorithm known as [Proximal Policy Optimization](#). Agent training is initiated from random parameters and does not bootstrap from replay of human-sourced training data. Agents are trained to maximize cumulative rewards related to such [game metrics](#) as net worth, kills, deaths, assists, and last hits. An AI agent is trained to maximize the exponentially decayed sum of future rewards with agents gain long-term rewards such as strategic map control while sacrificing short-term rewards. Cross-agent AI teamwork is controlled by a hyperparameter called "team spirit," which ranges from 0 to 1, putting a weight on how much each of OpenAI Five's heroes should care about its individual reward function versus the average of the team's reward functions.

The necessary next step: a broader transfer learning framework

As a next step from these types of AI-gaming testbeds, I would like to see a broader transfer learning framework that ties lessons learned to some larger [game-theoretic](#) framework. After all, game theory is a well-established branch of mathematics, is now heavily used in business and the computational social sciences, and is even being applied in life-science domains such as biology.

Why gaming AI won't help make AI work in the real world—but could

James Kobielus

The underlying game-theory principles could be applied to the real world, only if the test cases reflected the real world, not artificial fantasy environments

By

Contributor, InfoWorld | Aug 30, 2018

Why can't AI researchers frame their testbed project objectives to call out possible transferability of hard-won statistical knowledge in game-theoretic terms? For example, it would not be difficult to reframe the OpenAI Five implementation of DotA 2 in game-theoretic terms: cooperative vs. non-cooperative, symmetric vs. asymmetric, zero-sum vs. non-zero-sum, simultaneous vs. sequential ,perfect vs. imperfect information, brief vs. protracted, discrete vs. continuous, fixed vs. evolutionary, and so on. For guidance on to do that, check out how yours truly mapped online business-to-consumer engagement strategies to game-theoretic approaches in myblog post six years ago.

For a showcase on the power of AI-driven gaming in practical applications, I'd like to suggest an "AI safety" testbed focused on autonomous vehicles. It would transfer lessons from AI-driven gaming testbeds such as OpenAI Five, but with a keen focus on adaptive on-the-road strategic to improve vehicular safety.

When you think about it, the road network is a prime real-world example of the same core contexts as OpenAI Five. Getting from point A to point B is a high-stakes game in which we all participate all the time. Whether you're a driver, a passenger, or a pedestrian, it's clear that getting around is often a game of real-time continuous interactions over extended time horizons. It usually involves everybody dealing with partially observed states within high-dimensional action and observation spaces. Success on getting to your destination unscathed and on time depends on many-to-many interactions in which you and many other people execute never-ending sequences of moves, both strategic and inconsequential.

For example, this "courteous autonomous cars" research initiative could easily benefit from a gaming-like AI testbed. In fact, the researchers on this project have already begun to think through an approach that can easily be mapped to a testbed such as the one that OpenAI has built for geeky strategy games.

"Selfishness," the autonomous car researchers state, "has not been a problem with [autonomous vehicle algorithmic] approaches that predict human plans and react to them, because that led to conservative robots that always try to stay out of the way and let people do what they want. But, as we are switching to more recent approaches

that draw on the game-theoretic aspects of interaction, our cars are starting to become more aggressive. They cut people off, or inch forward at intersections to go first. While this behavior is good sometimes, we would not want to see it all the time."

If AI researchers can save lives by applying game-like approaches in the testing of alternative traffic management strategies, they can have a huge impact on making the world a better place.

James Kobielus is SiliconAngle Wikibon's lead analyst for AI, data science, and application development.