

Learning language by playing games

September 24 2015, by Larry Hardesty



Credit: Jose-Luis Olivares/MIT

MIT researchers have designed a computer system that learns how to play a text-based computer game with no prior assumptions about how language works. Although the system can't complete the game as a whole, its ability to complete sections of it suggests that, in some sense, it discovers the meanings of words during its training.

In 2011, professor of computer science and engineering Regina Barzilay and her students reported a system that learned to play a computer [game](#) called "Civilization" by analyzing the game manual. But in the new work, on which Barzilay is again a co-author, the machine-learning system has no direct access to the underlying "state" of the game program—the data the program is tracking and how it's being modified.

"When you play these games, every interaction is through text," says Karthik Narasimhan, an MIT [graduate student](#) in computer science and engineering and one of the new paper's two first authors. "For instance, you get the state of the game through text, and whatever you enter is also a command. It's not like a console with buttons. So you really need to understand the text to play these games, and you also have more variability in the types of actions you can take."

Narasimhan is joined on the paper by Barzilay, who's his thesis advisor, and by fellow first author Tejas Kulkarni, a graduate student in the group of Josh Tenenbaum, a professor in the Department of Brain and Cognitive Sciences. They presented the paper last week at the Empirical Methods in Natural Language Processing conference.

Gordian "not"

The researchers were particularly concerned with designing a system that could make inferences about syntax, which has been a perennial problem in the field of natural-language processing. Take negation, for example: In a text-based fantasy game, there's a world of difference between being told "you're hurt" and "you're not hurt." But a system that just relied on collections of keywords as a guide to action would miss that distinction.

So the researchers designed their own text-based [computer game](#) that, though very simple, tended to describe states of affairs using

troublesome syntactical constructions such as negation and conjunction. They also tested their system against a demonstration game built by the developers of Evennia, a game-creation toolkit. "A human could probably complete it in about 15 minutes," Kulkarni says.

To evaluate their system, the researchers compared its performance to that of two others, which use variants of a technique standard in the field of natural-language processing. The basic technique is called the "bag of words," in which a machine-learning algorithm bases its outputs on the co-occurrence of words. The variation, called the "bag of bigrams," which looks for the co-occurrence of two-word units.

On the Evennia game, the MIT researchers' system outperformed systems based on both bags of words and bags of bigrams. But on the homebrewed game, with its syntactical ambiguities, the difference in performance was even more dramatic. "What we created is adversarial, to actually test language understanding," Narasimhan says.

Deep learning

The MIT researchers used an approach to machine learning called deep learning, a revival of the concept of [neural networks](#), which was a staple of early artificial-intelligence research. Typically, a machine-learning system will begin with some assumptions about the data it's examining, to prevent wasted time on fruitless hypotheses. A natural-language-processing system could, for example, assume that some of the words it encounters will be negation words—though it has no idea which words those are.

Neural networks make no such assumptions. Instead, they derive a sense of direction from their organization into layers. Data are fed into an array of processing nodes in the bottom layer of the network, each of which modifies the data in a different way before passing it to the next

layer, which modifies it before passing it to the next layer, and so on. The output of the final layer is measured against some performance criterion, and then the process repeats, to see whether different modifications improve performance.

In their experiments, the researchers used two performance criteria. One was completion of a task—in the Evennia game, crossing a bridge without falling off, for instance. The other was maximization of a score that factored in several player attributes tracked by the game, such as "health points" and "magic points."

On both measures, the [deep-learning](#) system outperformed bags of words and bags of bigrams. Successfully completing the Evennia game, however, requires the player to remember a verbal description of an engraving encountered in one room and then, after navigating several intervening challenges, match it up with a different description of the same engraving in a different room. "We don't know how to do that at all," Kulkarni says.

"I think this paper is quite nice and that the general area of mapping [natural language](#) to actions is an interesting and important area," says Percy Liang, an assistant professor of [computer science](#) and statistics at Stanford University who was not involved in the work. "It would be interesting to see how far you can scale up these approaches to more complex domains."

More information: "Language Understanding for Text-based Games using Deep Reinforcement Learning."

people.csail.mit.edu/karthikn/pdfs/mud-play15.pdf

This story is republished courtesy of MIT News (web.mit.edu/newsoffice/), a popular site that covers news about MIT

research, innovation and teaching.

Provided by Massachusetts Institute of Technology

Citation: Learning language by playing games (2015, September 24) retrieved 28 April 2024
from <https://phys.org/news/2015-09-language-games.html>

This document is subject to copyright. Apart from any fair dealing for the purpose of private study or research, no part may be reproduced without the written permission. The content is provided for information purposes only.