Brentley Thomason  
CS 370 Project Two

Professor Mike Prasad  
June 19, 2025

Humans and machines go about solving problems in very different ways. When a person tries to solve a maze, they use their eyes to look around, spot patterns, and remember where they’ve been. They might notice dead ends or paths that look promising and use a simple method like always keeping their hand on the right wall. People can change their plan if they see something new, like a loop or a big open space, and can figure things out as they go.

On the other hand, a machine—like our pirate agent—doesn’t see or recognize things the way we do. It doesn’t “understand” the maze visually. Instead, it learns through numbers. It tries different moves and gets feedback in the form of rewards or penalties. Over time, and after many tries, it learns which actions are good and which ones aren’t. The pirate uses a type of learning called reinforcement learning (specifically deep Q-learning), where it slowly figures out the best moves to reach the treasure by learning from experience.

If a person were solving the maze, they’d look at where they’re starting and where the treasure is. They might choose a simple plan like always turning right. When they hit a dead end, they’d turn around and try a new path, remembering not to go back to the same spot. They might also just trust their guts and take shortcuts if something looks right.

The pirate agent starts in a random spot and looks at where it is in the maze (in a way that’s just numbers, not pictures). It decides what to do next either by trying something random or by picking what it currently thinks is the best move. After it moves, it gets a reward (good or bad), remembers what happened, and keeps learning. It repeats this until it either finds the treasure or fails, then it starts over and tries again.

Both the human and the pirate are trying to reach the treasure, but they go about it differently. A person remembers where they’ve been and learns fast, sometimes after just one try. The pirate needs many tries and lots of practice. A person uses their eyes, common sense, and simple rules, while the pirate relies on trying random moves and seeing what works. People can plan and adjust quickly, while the pirate improves slowly over time as it gathers more experience.

The point of the pirate agent is to learn the best set of moves to get to the treasure, even in tricky mazes were writing out all the rules would be too hard. The pirate learns by doing, not by following prewritten instructions.

When we talk about exploration, we mean the pirate tries new moves it hasn’t tried much before. Exploitation means it picks the move that has worked best so far. For this maze, it’s good to have a mix: first, the pirate should explore a fair amount, maybe 10% to 20% of the time. As it learns more, it should explore less—around 5%—so it can focus on the moves it knows work well. This way, it doesn’t get stuck doing the same thing over and over if there’s a better path.

Reinforcement learning helps the pirate figure out which paths lead to the treasure by rewarding good moves and punishing bad ones. Over time, it learns to choose actions that are more likely to lead to success. After enough practice, the pirate can find the best path without anyone telling it what to do.

To build this learning pirate, I set up a neural network that takes in information about the maze and gives back scores for each possible move. The pirate remembers its past experiences and trains the network using small batches of these memories. It uses a mix of exploring new moves and sticking with what it knows works (this is called ε-greedy strategy). The training ends once the pirate can reliably find the treasure no matter where it starts.

References

Sutton, R. S., & Barto, A. G. (2018). Reinforcement learning: An introduction (2nd ed.). MIT Press.

François-Lavet, V., Henderson, P., Islam, R., Bellemare, M. G., & Pineau, J. (2018). An introduction to deep reinforcement learning. Foundations and Trends® in Machine Learning, 11(3-4), 219-354. https://doi.org/10.1561/2200000071