Darren Sheftic

CS-370 Emerging Trends in CS

Prof. Binyam Heyi

8/15/2024

Design Defense

When a human tries to solve a maze they usually rely on their memory, intuition, and reasoning. A person might remember which paths they’ve already taken and if they hit a dead end they can backtrack to try a different direction. They might use logic and reasoning to avoid paths that seem like they’ll lead to nowhere (visible dead ends). Humans are good at making guesses and adjusting their strategy as they go along.

On the other hand, the pirate agent works very differently. Instead of relying on intuition or memory, it uses a process called Q-learning. At first, the agent doesn’t know anything about the maze, it has to explore by trying out different paths and learning from what happens. If it makes a good move, like getting closer to the treasure, it gets a reward. If it makes a bad move, like hitting a wall, it gets a penalty. Over time the agent starts to understand which paths are better because it learns from these consequences.

The main difference is that while humans can use their instincts and past experiences to solve the maze, the pirate agent needs to learn everything from scratch through trial and error. Both humans and machines might explore different paths, but humans do so more naturally, while the machine needs to learn and adapt over time.

The purpose of our pirate agent is to find the treasure in the maze all by itself. In technical terms, we’re asking the agent to solve a pathfinding problem, which means finding the best route to reach a specific goal.

For the agent to be successful it needs to balance two strategies, these are exploration and exploitation. Exploration is when the agent tries out new paths it hasn’t taken before. This is important because if the agent only sticks to what it knows, it might miss a shortcut or a better route. Exploitation is when the agent uses the paths it already knows works well. This helps the agent avoid making unnecessary mistakes.

The right mix of exploration and exploitation is key to finding the best path. If the agent explores too much, it might waste time on bad paths. If it exploits too much, it might get stuck in a routine and miss out on better options. Finding the right balance is important to helping the pirate find the treasure quick and efficiently.

We used a method called deep Q-learning to teach the pirate agent how to find the treasure, this allows the agent to learn from its actions. Every time the agent moves in the maze, it gets feedback in the form of rewards or penalties with the goal of the algorithm being to help the agent figure out which actions are the best in any given situation.

To do this we set up a neural network, which is like a mini-brain for the agent. This neural network helps the agent predict what will happen if it takes a certain action. Over time the agent uses this information to make smarter decisions to get to the treasure faster.

In practice, this algorithm works really well for complex problems like navigating a maze because it allows the agent to learn from its mistakes. Even though it might take some time for the agent to get good at finding the treasure, once it does it becomes very efficient at its task. The deep Q-learning method was a good choice because it helps the agent learn the best path through trial and error, which is exactly what we needed.