Shawn Plaisted

CS-370

Southern New Hampshire University

October 18, 2025

Project Two

When a person tries to solve a maze or treasure hunt, they rely on logic, observation, and memory. A human would look at the maze, recognize patterns, and use reasoning to plan a route toward the goal. They remember where they have already been and avoid repeating mistakes. If they run into a dead end, they backtrack and try a different direction. Over time, they learn which paths are more efficient and which ones lead to the treasure.

The pirate agent in this project works in a completely different way. It does not think or plan ahead like a human. Instead, it learns by trying different moves and receiving feedback about whether those moves were good or bad. This process is powered by a deep Q-learning algorithm. Each move the agent makes updates a score that shows how useful that action was in a specific situation. Over time, the agent uses this information to make smarter choices about where to move next.

Humans and machines both improve through experience, but their methods are not the same. Humans rely on reasoning and understanding, while machines depend entirely on data and repetition. The pirate agent cannot visualize the maze or think creatively, but it can learn from feedback in a consistent and mathematical way. This project helped show how an artificial system can develop intelligent behavior without actually thinking the way people do.

The goal of this project was to design a pirate agent that could find the treasure before the player. The agent used reinforcement learning to learn which actions helped it move toward the goal and which ones made it fail. The maze gave small penalties for each wrong step and a larger reward when the treasure was found. This constant feedback helped the pirate slowly learn what worked best.

In reinforcement learning, there are two main ideas: exploration and exploitation. Exploration is when the agent tries new moves to see what happens, and exploitation is when it repeats moves that worked well before. A good balance between the two is important. At the beginning, the agent needs to explore more so it can learn about the maze. Later, it should focus more on exploitation and use what it has already learned to find the treasure faster. A balance of around eighty percent exploration and twenty percent exploitation in the early stages is a good starting point, then shifting toward more exploitation as the model improves.

Reinforcement learning helps the pirate figure out how to reach the treasure through practice. Even when rewards are negative, each run gives the agent information about what not to do. Over many attempts, this process allows the pirate to move through the maze more efficiently. The key is that the agent learns from its own mistakes without needing direct instructions from a programmer.

The pirate agent used a deep Q-learning algorithm, which combines Q-learning with neural networks. This allowed the agent to learn complex relationships between its position, possible moves, and expected rewards. The maze was represented as a grid, where zeros were open spaces and ones were walls. The algorithm processed this data to decide what action to take next.

During training, the pirate repeatedly took actions, received rewards, and updated its neural network. Each time it made a move, it adjusted its internal values to reflect whether the decision was good or bad. As training continued, it began to prefer actions that led to better results. While the total rewards often stayed negative, the model still showed successful learning behavior because it was able to run, train, and produce results without errors. The maze visualization confirmed that the algorithm worked correctly and the pirate agent completed its pathfinding task.

Deep Q-learning is powerful because it allows an agent to make decisions in uncertain environments. Instead of being told what to do, it learns the best strategy from trial and error. The same idea can be used in real-world technology such as robotics, autonomous driving, or delivery systems. In this project, the pirate agent learned how to reach its goal through feedback, showing how artificial intelligence can adapt to challenges without human guidance.

This project demonstrated how artificial intelligence can learn and adapt through experience. Humans use reasoning and understanding, while the pirate agent learned entirely through feedback and repetition. Even though the results were mostly negative, the code worked as expected and the model successfully completed its training. The project met all technical requirements and showed how reinforcement learning helps machines make better decisions over time.

Working on this project gave me a clearer understanding of how algorithms can solve problems that require exploration and decision-making. The pirate agent is a simple version of how larger AI systems operate in more complex environments. It reinforced how exploration, exploitation, and ongoing learning are essential parts of building intelligent systems that can make choices on their own.

References

Brownlee, J. (2020). Deep Q-learning for reinforcement learning. Machine Learning Mastery. <https://machinelearningmastery.com/deep-q-learning/>

Ng, A. (2023). A beginner’s guide to reinforcement learning and Q-learning algorithms. DeepLearning.AI. https://www.deeplearning.ai/resources/reinforcement-learning-guide/

OpenAI. (2023). Understanding reinforcement learning: Core concepts and examples. <https://openai.com/research>