SNHU Capstone Project (Pirate Agent AI)

Noah Khomer

Southern New Hampshire University

CS 499 – Capstone Project

9/27/2025

SNHU Capstone Project (Pirate Agent AI) Part 2

For enhancement #2, algorithms and data structure, I have selected a different artifact. However, upon reading the submission and requirements based on the rubric, I ended up going back to the Computer Science 370 Pirate Agent project.

I created this. Project last semester where I was supposed to apply reinforcement learning to teach an agent how to navigate the maze so that it can find the treasure. The foundation of the artifact was a Q-learning implementation that was using an epsilon-greedy strategy and a Q-table to balance exploration and exploitation.

I ended up choosing this artifact again because it highlights both algorithmic problem solving and data structure optimization, both of which I felt are the core things required for this computer science project. There is no project. Implemented a functional reinforcement learning model, but it lacked a lot of advanced techniques and optimizations. Therefore, the enhancement I made demonstrates my ability.

Therefore, to apply an even more advanced reinforcement learning algorithm, such as double DQN. I ended up improving the efficiency for the data structure optimization using a prioritized experience replay. The third step I took was that I introduced caching so that there is significant performance improvement. I ended up using the Q-value caching systems to achieve this. Lastly, I had to communicate the results through professional-quality visualization, which was an analytics dashboard as used by so many agencies.

These changes made my artifact a strong representation of my growth in algorithms and data structures. While also demonstrating that I have industry-relevant skills in optimization and professional presentation by including files like README. Because of my enhancements, I was also able to achieve certain things.

For example, with the double DQN algorithm, I was able to separate the action selection and evaluation, which helped reduce overestimation of bias and stabilized the learning. Secondly, by using a prioritized experience replay, I was able to let. The learning speed focused updates. Independently.

In module one, I planned to enhance a different artifact, but it felt like I have. Worked on this artifact so much that I understood how to make it better with the existing code base. The most significant challenge that I ended up facing was debugging interaction between the new components, especially because I was using prioritized sampling with a double DQN logic to work together consistently.

Through this process, I ended up learning a lot of advanced algorithmic improvements. And I also learned that these algorithms require rigorous testing and careful consideration for the edge cases. Therefore, in conclusion, I can say that this artifact now demonstrates both my technical ability to enhance algorithms and my ability to optimize data structures to document and visualize results.

These enhancements may require an already good pirate agent project and an excellent fit for my e-portfolio. It highlights my growth as a computer scientist, and I am ready to apply these skills in the real-world environment.

References

Kahn, C. E. (2022). Hitting the mark: Reducing bias in AI systems. *Radiology: Artificial Intelligence*, *4*(5). [https://doi.org/10.1148/ryai.220171](https://doi.org/10.1148/ryai.220171%20)

Sewak, M. (2019). Deep Q Network (DQN), double DQN, and dueling DQN. *Deep Reinforcement Learning*, 95–108. [https://doi.org/10.1007/978-981-13-8285-7\_8](https://doi.org/10.1007/978-981-13-8285-7_8%20)

Song, J., Song, H., & Choi, W. (2015). Optimal caching placement of caching system with helpers. *2015 IEEE International Conference on Communications (ICC)*, 1825–1830. [https://doi.org/10.1109/icc.2015.7248590](https://doi.org/10.1109/icc.2015.7248590%20)

Zigon, B., & Song, F. (2023). Accelerating experience replay for deep Q-networks with reduced target computation. *Computer Science and Machine Learning Trends 2023*, 01–13. [https://doi.org/10.5121/csit.2023.130101](https://doi.org/10.5121/csit.2023.130101%20)