**4-2 Milestone Three: Enhancement Two: Algorithms and Data Structures**

Braedan French

Department of Computer Science, Southern New Hampshire University

CS-499-19469: Computer Science Capstone

Neil Kalinowski

June 1, 2025

**Narrative**

**Briefly describe the artifact. What is it? When was it created?**

The artifact chosen for the second enhancement originates from the CS-370: Current/Emerging Trends in Computer Science course. The artifact itself is a reinforcement learning algorithm based on deep q-learning that teaches an intelligent agent to seek the ending to a maze within a game. The program utilized two different Python files that established the rules, rewards, and structure of the maze for the game. Then a Python Notebook file with the code necessary to run training operations so that the agent could learn the game from scratch until it can win with 100% certainty.

**Justify the inclusion of the artifact in your ePortfolio. Why did you select this item? What specific components of the artifact showcase your skills and abilities in algorithms and data structure? How was the artifact improved?**

I selected this artifact for enhancement because I was not satisfied with the original results from the reinforcement learning algorithm. The algorithm did achieve success as the agent was able to achieve a 100%-win rate after hundreds of epochs and about 46 minutes of play time. But I am certain that the algorithm could be improved to reduce the amount of training needed for the agent to produce the same results. Hence why the artifact has been selected for enhancement within this category. As for the specific components that showcase my skills, the incorporation of a reinforcement learning algorithm shows the ability to design and implement algorithm solutions. Improving upon the learning rate of the intelligent agent through the reinforcement learning algorithm shows the ability to improve algorithm efficiency, and effective manipulation of algorithm hyperparameters. The current model used for the reinforcement learning algorithm is model-free, showcasing comprehension of algorithm models and how to establish them. The current improvement to the artifact is through manipulation of hyperparameters to increase the learning speed of the agent to lower the number of epochs and play time needed to achieve 100%-win rate. At the time of writing, I am still experimenting with combinations of hyperparameters as most combinations have been on par or worse than the original algorithm hyperparameters.

**Did you meet the course outcomes you planned to meet with this enhancement in Module One? Do you have any updates to your outcome-coverage plans?**

The course outcome I outlined to be covered by this enhancement was the third outcome. This outcome is to design and evaluate computing solutions that solve a given problem using algorithmic principles and computer science practices and standards appropriate to its solution while managing the trade-offs involved in design choices. Given that the original problem of achieving 100%-win rate through the reinforcement learning algorithm has been achieved, I have shown my ability to design solutions through algorithmic principles while managing trade-offs. This is because reinforcement learning algorithms themselves are a balancing act of managing hyperparameters and choosing the correct model for the problem. Each aspect greatly affects the learning speed of an intelligent agent, resulting in much longer training if improperly handled. As mentioned previously, progress on the enhancement has been slow with many combinations of hyperparameters being explored. Currently, the enhancement itself has yet to achieve this course outcome since the optimal combination of hyperparameters has not been determined. I will continue to experiment as results thus far have not produced success in increasing learning speed of the intelligent agent. And for updates to my coverage plans, there are no current updates to mention as I am still trying to achieve both the outcomes listed for this enhancement and the one planned for the previous enhancement.

**Reflect on the process of enhancing and modifying the artifact. What did you learn as you were creating it and improving it? What challenges did you face?**

I was aware how sensitive a reinforcement learning algorithm can be to the manipulation of hyperparameters, but now I have seen it firsthand. I have conducted tests with many different values for epochs and batch size to compare with the original results. And as hard as it is to believe, there has not even been a slight improvement to the learning rate of the intelligent agent. The experience of enhancing this artifact has really taught me how much effort it can be to improve algorithm efficiency with even something as simple as changing a few variable values. At this stage of enhancement, testing different hyperparameter combinations has been the biggest challenge due to the time required for each test. The original settings of the algorithm achieved 100%-win rate in about 46 minutes, meaning that each test is to be given just as much to show improvement. It is a lengthy process that so far has not yielded any success, and I worry that nothing I do will change that. If it continues like this, I will likely have to switch to my other plan for enhancement. Which is to determine the feasibility of incorporating a model-based algorithm instead of the existing model-free one. One other small roadblock in the process of enhancement was installing the correct packages for the program to run properly. Beforehand, the program was set to run in a digital environment that was preloaded with all the necessary packages. So, I had to determine which ones were needed within my own Python environment and get them installed to continue enhancement. After googling around, I was able to get everything sorted out.