Project 2 Defense

Paul Lamy

CS370

17 April 2024

Professor Jason Richard

In video games, non-player characters (NPCs) are in-game characters that are non-playable and made to interact with the player. Making an NPC that can think at a high level is important for an enjoyable gaming experience because it enhances the gameplay dynamic and makes a more realistic environment. As the AI developer for this game company, I wanted to make a great NPC powered by AI. This document explores how the pirate intelligent agent uses AI.

While it may seem obvious that humans and machines solve problems in many different ways, there are some similarities. As far as differences, while a human would use spatial reasoning and memory of past experiences to learn things, a machine has to use algorithms and predefined instructions. But both humans and machines learn from experiences. For a human agent to complete the maze, the human would walk in some direction until a wall is reached, choose what direction to move in, and then move in a new direction. Those 3 steps will be completed until a dead end is reached. A human would also know when to turn around, backtrace, and when to go in a new direction. When the intelligent agent in the Q-learning example is going through the maze, a direction will be chosen, and the agent will go in that direction. In using Q-learning, what makes the intelligent agent different is that the intelligent agent utilizes a Q table, which allows it to map every state to the amount of reward attained at each step (Schoberg, 2020). The human agent uses past experiences on which paths were not optimal to take, while the intelligent agent does the same but measures the rewards to find the path that gives it the highest reward.

When discussing Q-learning in this context, the distinction between exploitation and exploration must be made. Exploitation refers to the agent trying to seeking higher rewards, whereas exploration is searching through more of the state space, in this case the maze (Schoberg, 2020). For this case, while both are equally important, exploration should be utilized more. This is for two reasons: first, exploitation revolves around finding the highest reward, which is not the main goal of the maze exploration; second, the maze exercise is about finding the path to the end of the maze, not about finding the highest rewards along the way. For the pirate to find the path to the end of the maze, reinforcement learning can be utilized by having the pirate traverse to a certain point, calculate the reward it has acquired thus far, and then learn from its current progress in that epoch to find the next best path. This is known as trial and error, where the pirate agent explores different actions and analyzing their consequences.

In this game, I utilized Q-learning to make a pirate intelligent agent that could traverse the maze. The Q-learning algorithm I made runs for a certain number of epochs, which can be set by the user. Then, for each epoch, an agent cell is randomly selected, the current state is observed, and the game starts. When the game starts, either exploitation or exploration is done depending on a randomly generated output. The experience of the algorithm, in this case the pirate intelligent agent, is remembered by the agent. After that, the neural network model is trained on the data and loss is evaluated. At the end of that iteration, the win history and win rate are both calculated. Overall, this Q-learning algorithm can use either exploitation or exploration until the game is over, then record the results, and use those results to train the neural network to improve the pirate intelligent agent. This makes for a more intelligent agent that can solve the maze on its own.