Marcus Doucette

2025-06-22

CS370

Project 2 Design Defense

* **Analyze the differences between human and machine approaches to solving problems.**
  + Describe the steps a human being would take to solve this maze.
    - A human would look at the entire board and first need to work to understand the rules. Once they understand the rules they will likely look for heuristics to form a plan around. In this case it might be generalizing the map into areas like rooms and then planning out moving from one to the other. Lastly the human would need to execute this plan using the controls of the game first moving the agent down into the first room then right and up into the second and so forth until the agent gets to the exit. This process of solving separate problems in a sensible order to make each stage easier is core to how humans solve so many problems so effectively.
  + Describe the steps your intelligent agent is taking to solve this pathfinding problem.
    - The intelligent agent guesses at which actions will get it closer to it’s goal, and each action has an associated reward. These actions and there rewards are then stored in a table along with the state of the board before the action was taken. This process lets the agent build up a working memory of the different states the board could be in and what to do in those states. Each state of the board then produces a q table with a predicted reward for each possible action, and by having a sufficiently developed table, the best reward at each stage will move the agent closer and closer to the end until it wins. Computers using deep q learning use their large memory sizes to store information about all possible states the problem could be in and the training then informs which actions should be taken in those states to get closer to the goal.
  + What are the similarities and differences between these two approaches?
    - Both the human and the AI need to interact with the game in very similar ways but the human conceives of it as a discrete set of problems that can each be solved easily, while the AI uses a large memory to understand all possible states of the game and be able to optimize for the best move in each of the states.
* **Assess the purpose of the intelligent agent in pathfinding.**
  + What is the difference between exploitation and exploration? What is the ideal proportion of exploitation and exploration for this pathfinding problem? Explain your reasoning.
    - Exploitation is using what you already know in order to achieve an outcome. Exploration is doing something you don’t know to learn. I think that this agent would need to explore until it has a path it can exploit to the goal, the real problem is creating this understanding and then being able to optimize it to be more effective. This pattern (while not the direct result of deep q learning) would help the AI use the way that it think’s best to it’s fullest in learning when it doesn’t know how to win and then using what it already knows to win quickly when it can see a way to do so.
  + How can reinforcement learning help to determine the path to the goal (the treasure) by the agent (the pirate)?
    - Reinforcement learning focuses on more of a heat map style approach where the agent will ultimately view spaces that are closer to the goal as more valuable and then just move towards the valuable spaces every turn. If the heat map was generated correctly then this should always succeed, and demonstrates a good constant balance of exploration and exploitation.
* **Evaluate the use of algorithms to solve complex problems.**
  + How did you implement deep Q-learning using neural networks for this game?
    - My biggest problem with this project is that I don’t feel like I really did implement the deep q-learning neural network instead needing to rely on keras to do most of the work for me, which made things needlessly difficult when keras wasn’t cooperating. At least in theory, the neural network would take in the state of the board and then produce a table with the value of doing any of the options that would then functionally be similar to the output of the Q-table. The reinforcement rewards were something that was implemented in code I wasn’t supposed to touch so messing with these details was more difficult than I believe it ought to have been.