CSD311
Artificial Intelligence
Classic Snake Game

Prahansha Kumaravel	2110110376
Saransh Saxena	2110110465
Tarang Verma	2110110550
Mansi	2110110623
Abhiveer Singh	2110110820
Pragunjay Singh	2110110874

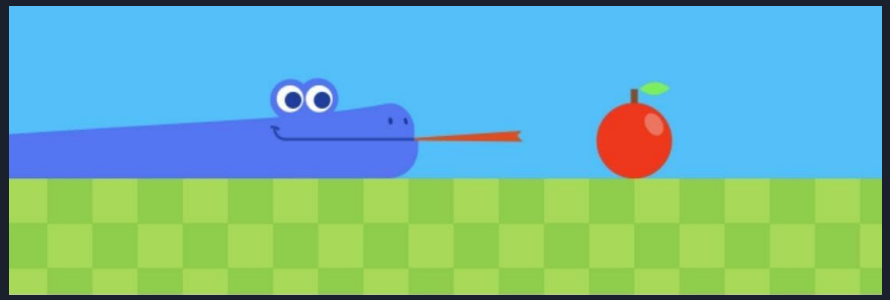
Contents

Overview

The Snake game is a 2D grid-based challenge where players guide a snake to consume food while avoiding obstacles and collisions. This project leverages AI to automate gameplay, employing techniques such as heuristic search and reinforcement learning to enhance the bot's intelligence, decision-making, and performance. The goal is to optimize the snake's movements and adapt effectively to different game scenarios.

Problem Analysis:

- Game Dimensions:
  - Height - 480 pixels = 24 blocks
  - Width - 640 pixels = 32 blocks
- The snake must go to and eat the apple, which will increase its length by 1 block.
- The apple is produced randomly on the grid on any available empty space.
- The goal is to get the highest score possible.
- The snake can move up, down, left and right, and can only make right angle turns.
- The game will end if:
  - The snake collides with the walls of the game
  - The snake collides with its own body



Teaching AI how to play Snake Game

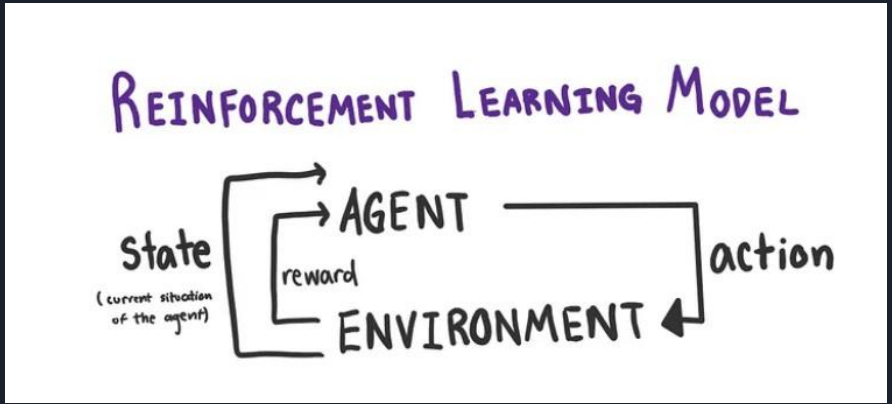
Algorithm Overview

- Greedy Best-First Search:
  - Finds shortest path using Manhattan distance.
  - Potential collisions, leading to suboptimal results.
- A* Search:
  - Considers both path cost and heuristic distance.
  - Balances efficiency and safety.

Reinforcement Learning

We are using RL algorithm to instruct a software agent on appropriate behavior within an environment by evaluating its performance.

Deep Q Learning approach in RL extends its capabilities by incorporating a deep neural network to predict and anticipate actions. The integration of deep learning enhances the agent's ability to make informed decisions



Reinforcement Learning

State Representation: The state of the game, a critical input for the agent, comprises **11 boolean values** reflecting various aspects, including dangers in different directions and the relative position of food.

[danger straight, danger right, danger left,

direction left, direction right, direction up, direction down,

food left, food right, food up, food down]

Reinforcement Learning

In the training process, the agent interacts with the game environment. The agent's state is obtained from the game, an action is determined based on the state, and the model predicts the action. Subsequently, the agent plays a step in the game, receiving rewards, information on the game state, and the score. This information is then used to train and update the model.

Reward and Action:

Rewards:
Eat food: + 10
Game over: − 10
Else:0

Action:
[1, 0, 0] → straight
[0, 1, 0] → right turn
[0, 0, 1] → left turn

Reinforcement Learning

The game is played thousands of times. The agent learns by updating its Q-values using the Bellman Equation:

$$Q(s,a) = r + \gamma \max(a) Q(s', a')$$

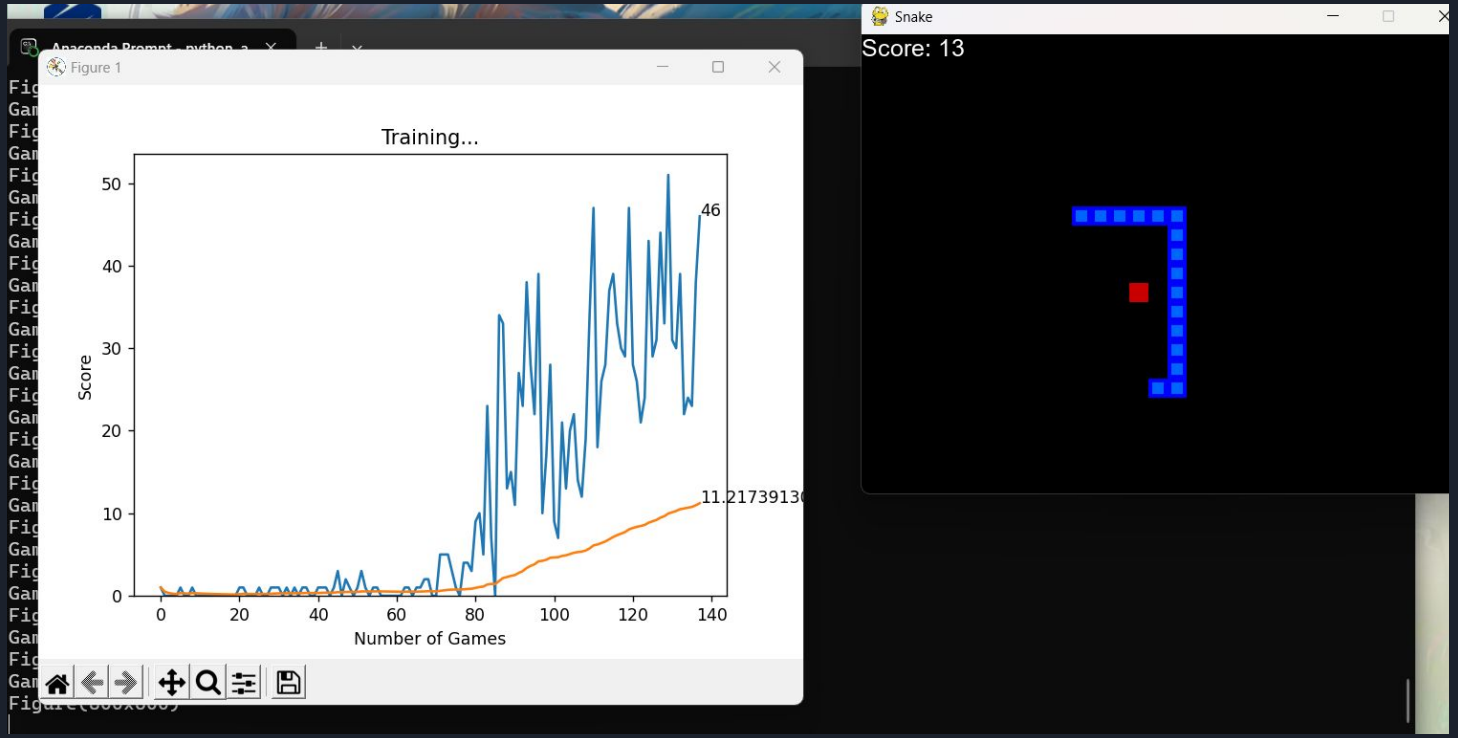
- s: Current state
- a: Current action
- r: Reward for the action
- s' : Next state
- a': Next action
- γ: Discount factor for future rewards.

Reinforcement Learning

Deep Q Learning: The Q value, signifying the quality of an action, is iteratively updated. This involves considering the current Q value, the reward, and the maximum expected future reward given a new state and all possible actions.

1. Initialize Q Value (= init model)
2. Choose action (model.predict(state)) (or random move)
3. Perform action
4. Measure reward
5. Update Q value (+ train model)

Final Result



Observations

Reinforcement Learning is the best algorithm to teach AI how to play Snake Game. While it is slow in the beginning, as the number of games increases, the AI learns how to play the game correctly, and is able to navigate itself perfectly.

- Greedy algorithms are efficient but lack foresight.
- A* balances cost and heuristic for strategic moves.

Conclusion

This project successfully demonstrated the application of Reinforcement Learning, specifically the Deep Q-Network (DQN) algorithm, to train an AI agent to play the classic Snake game. The agent learned optimal strategies through trial and error, maximizing its score over time. This project highlights the potential of AI in mastering complex tasks and offers insights into the capabilities of Reinforcement Learning techniques.

THANK YOU