

The Game of Snake

Anta Fall

What to Expect

In this project, the AI will learn how to play the game of Snake. Utilizing reinforcement learning an RCNN's.

$$\underbrace{NewQ(s, a)}_{\text{New Q value for that state and that action}} = \underbrace{Q(s, a)}_{\text{Current Q value}} + \underbrace{\alpha}_{\text{Learning Rate}} [\underbrace{R(s, a)}_{\text{Reward for taking that action at that state}} + \underbrace{\gamma}_{\text{Discount rate}} \underbrace{\max Q'(s', a')}_{\text{Maximum expected future reward given the new s' and all possible actions at that new state}} - Q(s, a)]$$

What is Reinforcement Learning?

Reinforcement Learning is how software/machine learning agents learn to take action in an environment in order to maximize its award.

“Reinforcement learning is an area of machine learning concerned with how intelligent agents ought to take actions in an environment in order to maximize the notion of cumulative reward. Reinforcement learning is one of three basic machine learning paradigms, alongside supervised learning and unsupervised learning. ” - Wikipedia

Why Reinforcement Learning

- ❖ Compared to other deep learning models such as LSTM, and RCNNs and such. Reinforcement learning helps to train machine learning models to make a sequence of decisions.
- ❖ In terms of LSTM, LSTM would allow for learning long-term dependencies.
- ❖ RCNN's (Region-based Convolutional Neural Networks) allow for object detection.

What is the Game of Snake?

Snake is a game where the user/snake maneuvers around the board, trying to collect an orb and as the snake continues to collect the orb it will grow bigger, making it harder to move around the board.

The objective for this project will be for the AI/machine learning module to gain the highest score, whilst trying to collect the orb as it moves at a random position across the board. Utilizing RCNN's object detection, the AI will be able to collect the orb on its own as it learns from its environment.

However, it must do so without tangling itself - otherwise the game will be over, and the score will reset to zero. A score will be kept to ensure the AI is maximizing its goal.

Approach

These must be installed for the code to run properly

- The tools used to achieve this project will be the following:
 - PyCharm or any working Python IDE
 - PyGame: Helps to create the game UI.
 - PyTorch: Provides Machine Learning framework
 - NumPy: Helps with in-game arrays
 - Python version 3.7+
 - Testing on Repl.it: Online compiler and Python IDE

Code + Demo

References

- <https://towardsdatascience.com/various-ways-to-evaluate-a-machine-learning-models-performance-230449055f15>
- <https://medium.com/@sprhrlabs/understanding-deep-learning-dnn-rnn-lstm-cnn-and-r-cnn-6602ed94dbff>
- https://en.wikipedia.org/wiki/Reinforcement_learning
- [https://en.wikipedia.org/wiki/Snake_\(video_game_genre\)](https://en.wikipedia.org/wiki/Snake_(video_game_genre))