İTÜ

BLG 435E Artificial Intelligence - 2021/2022 Fall Assignment 2

December 12th, 2021

In this assignment, you need to design and train an agent with Deep Q-Network (DQN) algorithm for the Snake environment. Assignment includes two parts which are designing (1) state representation and (2) implementing+training DQN algorithm.

The environment is written in Python with PyGame. Like in homework 1, you need to install PyGame package. It is a classical game, hopefully, each one of you are familiar with. Snake environment is given in 3 different modes:

1. CLASSIC mode
2. TRON mode: In this mode, snake will grow up at every time step whether it eats the apple or not.
3. NOTAIL mode: In this mode, snake does not have a tail and will not grow up when it eats the apple.

1 Design of State Representation (50 pts)

You will use the information given from the environment and use this information for designing state representations for your agent. Your state representation can include any information you can get from the environment (snake position, snake direction, apple position, near wall condition, near apple condition etc. and if you need to, you can add helper functions to get the data you want from the environment). In your report, you should explain the reasons for your state representation choices for each environment mode.

You should design the reward functions similarly, such that there are 3 reward functions for 3 modes. You should give reasonable arguments for your selections in your report.

After the implementation and training (see Part 2), if you decided to change your representations, discuss the reasoning behind your decisions and analyse their effects to the results.

2 Implementation and Training of DQN (50 pts)

You are asked to implement a Neural Network for an existing DQN implementation. In the given implementation, a framework called PyTorch is used. You can install and learn more about PyTorch from here: https://pytorch.org/get-started/locally/. You are expected to create the network and train the agent for NOTAIL, CLASSIC and TRON environment modes successfully. The agent you trained should show some promise to be counted as successful.

During your trainings, try to find the ideal hyperparameters by testing different ones. Note that the learning rate is the most important hyperparameter. Experiment around with them by making educated guesses around the initial values given to you, eventually fine tuning them. Explain your important findings such as how the changes you made affected the training in your report. You can fine tune for a single environment mode and use the same hyperparameters for the rest.

Plot the reward and loss values for the hyperparameters you have found. In other words, during the training with those hyperparameters, plot the changes over time for reward and loss, again, for each environment mode. Analyse your results in your report.

You should run the environment for at least 25000 steps and if your machine handles, you should go above that.

3 Implementation Details

You need to fill the following functions in the given template files:

- In DQN.py: PolicyNetwork class
- In DQN.py: select_action function in DQN class
- In DQN.py: necessary areas in update function of DQN class
- In snake.py: In CustomSnake class, get_state and get_reward functions

You need to plot charts and put them in your report for displaying loss and reward outputs.

Remarks

- You need to get familiar with PyTorch and tensors. Follow the official tutorial https://pytorch.org/get-started/locally/ if you need.
- As a reference: 50k steps took 20 minutes with Ryzen5 1600 CPU. Arrange your time well.
- The given template is implemented with PyTorch version 1.4.0.

4 Submission

Submit your homework files through Ninova. Please zip and upload all your files using filename "BLG435E-HW-2-STUDENTID.zip". You are going to submit:

1. All your filled code files for the given code template.
2. A PDF file report including:
   a. Explanations about your state representation design
   b. Analyses about hyperparameter selections for your network design
   c. Instructions to compile/run your code

- Your code must be able to compile and run on ITU's Linux Servers without any errors, otherwise your code may not be evaluated.
- Pay attention to the deadline of the homework, including hour.
- In case of any questions, feel free to send an e-mail to Res. Asst. Gamze Akyol **akyolga@itu.edu.tr**.