Proposal

Kung-fu Master Atari Game Autonomous Play

Ιωαννίδης Χρήστος (2018030006)

- what problem/game/domain/challenge do I plan to study?

  There is an atari game called kung-fu Master also supported as an open-ai gym environment, i intend to automate it to play by itself OK!

  https://www.gymlibrary.dev/environments/atari/kung_fu_master/

- what characteristics does it have?

  A rather simple side-scroller game that the player moves towards a direction while small teams of enemies attack him. The player uses his Kung-fu moves to defeat them and continue. OK! I also see there are 8 actions and a huge state space (console image)!

- which algorithm am I thinking of implementing for my agent?

  I am thinking of implementing a Q-learning algorithm, however the statespace might be too big so I will probably have to change to deep Q-learning.
  For sure, tabular Q-learning is not enough, you have to apply some kind of appoximation!
  That being said, I need to find time to study for the exams and I'm already in a tough spot. The idea described above requires a great deal of effort and time to implement (also i haven't even thought about the training part much, i'm gonna use pytorch that can take advantage of the CUDA inside my laptops rtx-3050 4GB however in don't know if it will be enough, i will really dislike the idea of configuring a cloud computing space just for this project)
  You can certainly master learning for this game with such a laptop, unless you come up with a really deep NN!!
  In Case in see that i find the above implementation idea unfeasible due to time, effort or hardware constraints i'm going to turn to a simple behavioral model maybe using a heuristic for the score and just detecting enemies and hitting them based on proximity (something like the 1-2 project of this class)

- what kind of data or which simulator will I use?

  Open ai Gym OK! This give you a really good start! You just need to focus on the agent!

- which programming language or software package will I use?
  OK!
  Python of course, with pytorch,numpy and whatever else is needed in the way

- what similar or related work is there in the literature?
    1. **Some older projects of this class have used similar methods and technologies**
    2. **I have already implemented a Q-learning algorithm for the Reinforcement Learning class as an assignment https://github.com/kirisaki-momotaro/policy-iteration-Q-learning-implementations**

       Also for the AI class i have implemented as an assignment algorithms that are going to be of use in the behavioral algorithm case

       https://github.com/kirisaki-momotaro/tuc-ai-2022

    3. **Openai became headlines a fair amount of times for using deep Q-learning to play atari games**

OK! That means that you already have a fair amount of experience with RL.

- how does my proposal relate to the course?

  OF COURSE, it's Deep Q-learning, we have already seen Q-learning in class.

  Even the second option of the behavioral part will basically be a much more sophisticated implementation of this class 1-2 project so my guess is that its is related to the course.

  Good!

  Question: To what extent can i use readily available code in my project?? Copying a Deep Q-learning algorithm and adjusting it to work with the game is fine? Or should i write everything from scratch?

  You don't have to build everything from scratch! You can use existing code and/or libraries.

  the deadline will be 1-2 weeks after the exams.

  Also please brief us as soon as possible on this project's deadline and also about the 4th project so as to scale of our project accordingly. will do so soon!

Go ahead and get started!

Good luck!