Accelerating Deep Reinforcement Learning

Course Project
CSCE 790
(Machine Learning Systems)

Project description

- Deep reinforcement learning (RL) has achieved many recent successes.
- However, running experiments is a key bottleneck.
- The aim of this project is to utilize computer system capability (e.g., parallel execution) to accelerate training of Deep RL agents.

Selecting the environment

- Select 2 environments from OpenAI Gym environments: https://gym.openai.com/envs/#atari
- One of these environments must be Pong: https://gym.openai.com/envs/Pong-v0/

Selecting Deep RL algorithms

- Select 3 Deep RL algorithms
  - DQN: https://blog.openai.com/openai-baselines-dqn/
  - Ape-X DQN: https://arxiv.org/pdf/1803.00933.pdf
  - Double DQN: https://arxiv.org/pdf/1509.06461.pdf
  - Distributional DQN: https://arxiv.org/pdf/1707.06887.pdf
  - Prioritized DQN: https://arxiv.org/pdf/1511.05952.pdf
  - A3C: https://arxiv.org/pdf/1602.01783.pdf
  - A2C: https://blog.openai.com/baselines-acktr-a2c/
- Code: https://github.com/openai/baselines

Choose accelerating strategies

- Select 3 existing distributed approaches to accelerate training: e.g.:
  - Parallelized DQN: https://arxiv.org/pdf/1507.04296.pdf
  - Experience replay: https://arxiv.org/pdf/1803.00933.pdf
  - Distributed execution: https://arxiv.org/pdf/1802.01561.pdf
  - Parallelized A3C: https://arxiv.org/pdf/1611.06256.pdf
  - Increased batch sizes: https://arxiv.org/pdf/1705.04862.pdf
  - Multi CPU/ Multi GPU: https://arxiv.org/pdf/1803.02811.pdf

Deliverables

- You need to basically replicate existing results which are out there, however, if you have a new idea (e.g., using a stream processing pipeline to facilitate training) and if you show it is beneficial (you do not need to beat state-of-the-art!), you will get a perfect score.

- In this project, you will mainly spend time running experiments.

- In your final report, you will compare the results mainly in terms of "learning over time", i.e., Y-axis is score of the game and X-axis is time.

- Again how to best present your results is an art here, you may want to pay a special attention how the results are compared in the papers that I referenced in previous slides.

- Similarly to other projects, you need to deliver the actual experimental data as well as the report in terms of Python notebook.

Final remarks

- Use your creativity when it comes to analyzing the results, try to surprise me!
- If you find a very interesting observations and dig into it by providing some insight, you will then get a good score!
- If you also produce very good results, you may also want to think about a potential paper, it’s optional, but I strongly recommend it.
- Ray framework may be useful: https://github.com/ray-project/ray