BUILDING GAME-BOTS USING OPENAI’S GYM AND UNIVERSE

Anmol Jagetia
INTRODUCTION

- OpenAI - OpenAI is a non-profit AI research company, discovering and enacting the path to safe artificial general intelligence.
INTRODUCTION

• **Gym** - A toolkit for developing and comparing reinforcement learning algorithms.

• **Universe** - A software platform for measuring and training an AI's general intelligence across the world's supply of games, websites and other applications.
INTRODUCTION

• Reinforcement Learning - is a theory that allows an agent to learn a strategy so as to maximize a sum of cumulated (delayed) rewards from any given environment.
The two most important distinguishing features of Reinforcement Learning are:

- Trial-and-error search.
- Delayed reward.

Challenges faced by RL are:

- Trade-off between *exploration* and *exploitation*. 
INTRODUCTION

- GitHub - @anmoljagetia
- LinkedIn - anmoljagetia
- Twitter - @anmoljagetia
- Email - anmoljagetia@gmail.com
REQUIREMENTS

- Python.
- Golang.
- Docker.
- VNC Viewer.
- TensorFlow/Keras/Theano.
- OpenCV.
- tmux (the start script opens up a tmux session with multiple windows)
- htop (shown in one of the tmux windows)
- libjpeg-turbo (brew install libjpeg-turbo)
REQUIREMENTS

- six (for py2/3 compatibility)
- gym
- atari-py
- universe
- opencv-python
- numpy
- scipy
- jupyter
import gym
import universe  # register the universe environments

env = gym.make('flashgames.DuskDrive-v0')
env.configure(remotes=1)  # automatically creates a local docker container
observation_n = env.reset()

while True:
    action_n = [[[('KeyEvent', 'ArrowUp'), True]] for ob in observation_n]
    observation_n, reward_n, done_n, info = env.step(action_n)
    env.render()
import gym
import universe  # register the universe environments

env = gym.make('flashgames.DuskDrive-v0')
env.configure(remotes=1)  # automatically creates a local docker container
observation_n = env.reset()

while True:
    action_n = [[[('KeyEvent', 'ArrowUp', True)] for ob in observation_n]
    observation_n, reward_n, done_n, info = env.step(action_n)
    env.render()
Let’s Code!
OPENAI PROVIDES A STARTER AGENT WHICH IS STATE OF THE ART
DEMO!
At the start of the training.
After training for ~3 hours.
After training for ~30 hours.
Sometimes it learns interesting things!
When modelled with a “do nothing” state.
FUTURE IMPROVEMENTS

• Forza Motorsport 4 - Allows you to create an AI that learns how you drive and imitate your style!
ALTERNATIVES

Alternate Environments for rendering game bots:

- Arcade Learning Environment.
- PyGame Models.
- PyGame Learning Environment. (Gym has an interface, gym-ple).
- Unity Models.
ALTERNATIVES

Alternate Algorithms:

- NEAT.
- Tree search algorithms.
- DeepMind’s EWC.
THANK YOU!

The best way to predict the future is to invent it.

- Alan Kay