Deep Reinforcement Learning: Tic-Tac-Toe

Kevin Liang

Duke University

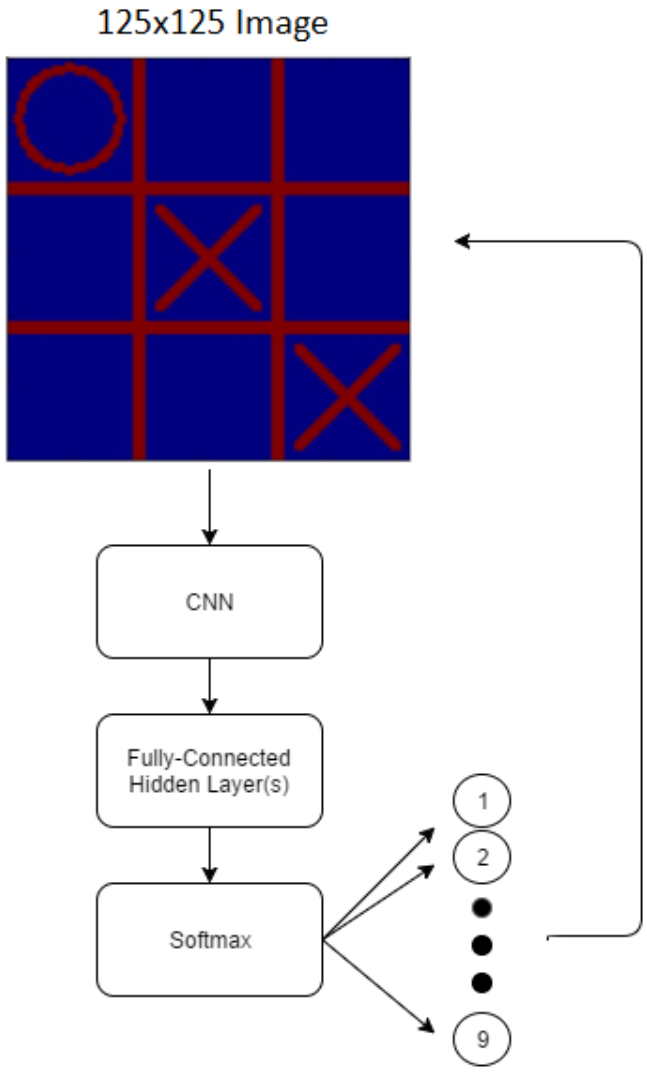
02 July 2016

Introduction

- Goal: Have an agent learn how to play Tic-tac-Toe using deep reinforcement learning
- Tic-Tac-Toe game
  - Players X and O take turns placing marks on a 3x3 grid
  - Player that achieves a 3-in-a-row wins
  - If the board fills up without a winner: draw
- Toy example to get started before tackling more complicated settings (eg. Atari games of OpenAI Gym)

Deep Reinforcement Learning Agent Set-up

- Agent is given a 125x125 image of the state of the board, not the 3x3 state space directly.
- Agent not informed of rules. Does not know:
  - what leads to a win/loss
  - own identity
  - marking an already occupied spot is an illegal move
- Perform weight updates using RL algorithm:
  - Policy Gradients
  - Q-Learning



Optimal opponent

- Newell and Simon Tic-Tac-Toe Rules:
  - Rule 1: If agent can win, then make a 3-in-a-row
  - Rule 2: If opponent can win, block the winning move
  - Rule 3: Make a fork (two 2-in-a-rows)
  - Rule 4: Block an opponent's fork, while simultaneously making a 2-in-a-row if possible
  - Rule 5: Take the center
  - Rule 6: If opponent has a corner, take the opposite corner
  - Rule 7: Take an empty corner
  - Rule 8: Take an empty side
- Add a "difficulty" parameter that controls how often the "optimal" agent makes a random move, instead of always following rules
  - Allows for the deep RL agent to occasionally win (and therefore learn)

Questions

- CNN architecture details
- Hyperparameters (CNN and RL)
- Computer/GPU/cluster to train it on