Capstone Project Proposal

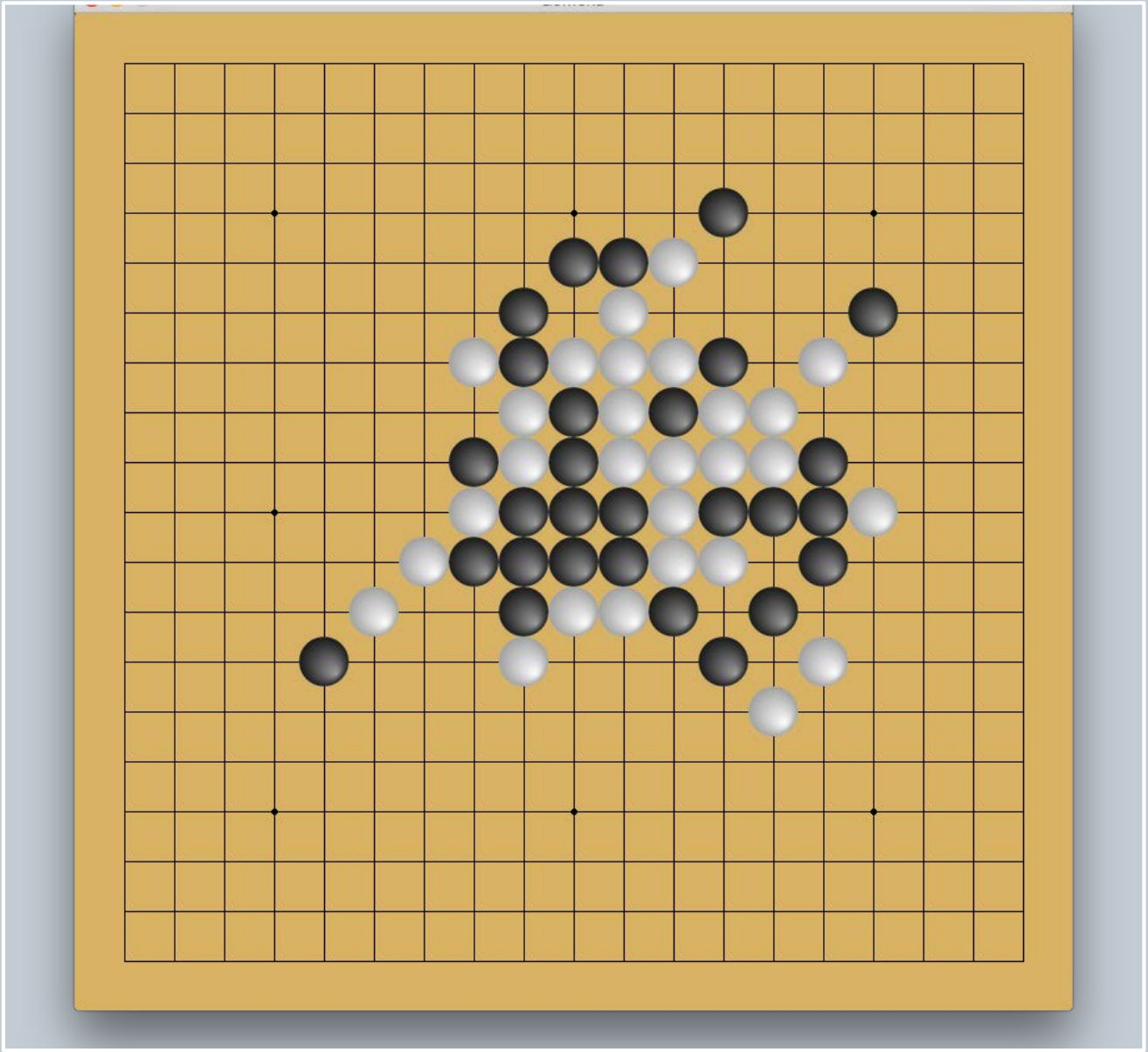
Using Reinforcement Learning to Train a Gomoku Playing Agent

Tong Zou

DOMAIN BACKGROUND

Gomoku is an abstract strategy board game. Also called Gobang or Five in a Row, it is traditionally played with Go pieces (black and white stones) on a go board with 19x19 (15x15) intersections; This game is known in several countries under different names.

Black plays first if white did not win in the previous game, and players alternate in placing a stone of their color on an empty intersection. The winner is the first player to get an unbroken row of five stones horizontally, vertically, or diagonally.



Problem Statement

- Gomoku can be really challenging to win even though the rules are easy. Gomoku is still new and it attracts many people to try to create a good AI for it, but it is nowhere near being totally solved.
- Our task is to make a Gomoku game AI which can learn the rules of Gomoku by playing games with itself, which is similar to the AlphaGo AI by Google Deep Mind.

Datasets and Inputs

- I will use OpenAI Gym to simulate the environment. Gym doesn't have a Gomoku environment yet, I will have to create the environment first.
- Training data will be recorded by OpenAI Gym. Training video can also be recorded.

Solution Statement

- Use OpenAI Gym as reinforcement framework.
- I plan to use Policy Gradient similar to AlphaGo. I will use a NN to handle the policy function, and use policy gradient method to train the NN.
- I will use a classic AI acting as the opponent and train my agent.

Benchmark Model

- There is a Gomocup which is a competition between Gomoku AI agents. But unfortunately it only supports windows and is mostly C/C++ based.
- I have found a couple python based AI agent. One is a more classical tree search/board evaluation kind of AI, and the other uses Q-Learning. I will compare my agent against these two.

Evaluation Metrics

- The trained agent will be evaluated against the benchmark model.
- It will also be evaluated by playing against itself with trained model at different stage, i.e., new models played against the old model and this should show whether our agent is learning or not.