

Title: Using AlphaZero to play chinese Chess

Team: Yifeng Liu, Jiahua You

Role: "AI as player" (e.g. "AI as progression designer")

Overview: As everyone is getting familiar with AlphaZero, we want to implement AlphaZero in chinese chess. Chinese chess is different than other type of chess. The policy network and value network combining with CNN are especially designed for Go. In Go, the goal of the game remains the same, to win the maximum area of spaces. It just fits with convolutional neural network. The whole board is symmetric. Each move just needs to place a single piece in one position. Result is also simple, win or lose. If we want to make alphaZero general to all chess, such like chinese chess, it needs rethink the process. First of all, different piece has different rule of action. The board is unsymmetric which will affects the moving direction. And there are even result(no wins, no lose). Traditional CNN may not fits perfectly in this situations. Moreover, there are countless of complete data in Go. Compared to chinese Chess, the only data is called final phase. One way to simplify training process is to give machine the incomplete data, and AlphaZero will trains himself to the final phase.

Key technical demo(s): In summary, the AlphaZero can be divided into two parts: MCTS and neural networks. In this project, we want to abandon human chess knowledge. Using self-playing to learn how to play chess. The process is like this: First, the chess manual is generated, then the chess manual is used as input to train the neural network, and the trained neural network is used to predict the game and the

winning rate. The AI will start self-playing with a whiteboard. He just knows the rule but he doesn't know the strategy. So every move is random. And each move is simulated by MCTS. So essentially, MCTS is used to maintain a tree, which saves the information of situation and action.

Novelty: It will inspect the generality of alphazero. The further research will be on how to transplant training model to mobile device.

Value: Who should be interested in this project (e.g., researchers in specific fields, practitioners in specific industry roles, etc.)?

What will they find valuable? It's not expected that they will literally reuse any code from your project, but they might take your project as inspiration. ("Teachers who have no programming experience should be able to use a system like mine to build their own adaptive progressions of educational content represented as rich-text Markdown documents.")

Building blocks: The system will be built on Python3, TensorFlow_gpu.

The chess engine will be modified by:

<https://github.com/Linzertorte/ChineseChess>