Playing Go without Game Tree Search Using Convolutional Neural Networks

Jeffrey Barratt

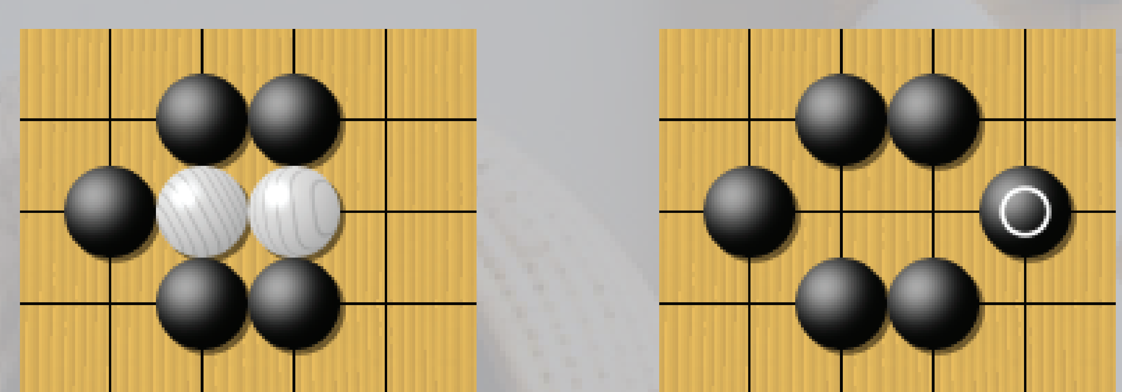
jbarratt@
cs.stanford.edu

Chuanbo Pan

chuanbo@
cs.stanford.edu

Introduction

A game of Go consists of two players alternating placing stones on a 19 by 19 board, with the player with black stones starting. Any stones that are directly adjacent to each other are part of the same group. If any group is completely surrounded by opponent stones, it is taken off of the board, as shown in Figure 1. The goal of the game is to surround as much territory as possible.



Before After

Figure 1: Black can capture the two white stones by playing at the circled point. The stones are then taken off the board.

Problem Statement

While the rules of Go are simple, mastering the game is not. Go has long been the hardest perfect information game to play well using computers. Previous and recent approaches such as AlphaGo and Crazy Stone all use a variant of a game tree search, which explores orders of magnitude more states than a human can. Our goal was to create a player that limits itself to only considering the current board position, and could thus attempt to mimick the deep human understanding of the game.

Dataset and Methods

Our approach involved three distinct methods and structures: Supervised Learning, Reinforcement Learning, and Cross-Shaped Convolutions.

Our data set consists of 53,000 professional games from as early as the 11th century to 2017 in Smart Game Format (SGF). Each of these games is processed into a feature vector and move pair that shows what move was made by the professional player in what game state.

This yielded over 10 million state-move pairs, which were used to train the network via supervied learning. We further trained the network through games of self play, where we would play it against previous iterations of the network to increase variety and decrease overfitting.

We also introduce a novel cross convolution shape (As seen in Figure 2) to deal with the importance of diagonal board positions to the game of Go. This allowed us to obtain greater classification accuracy.

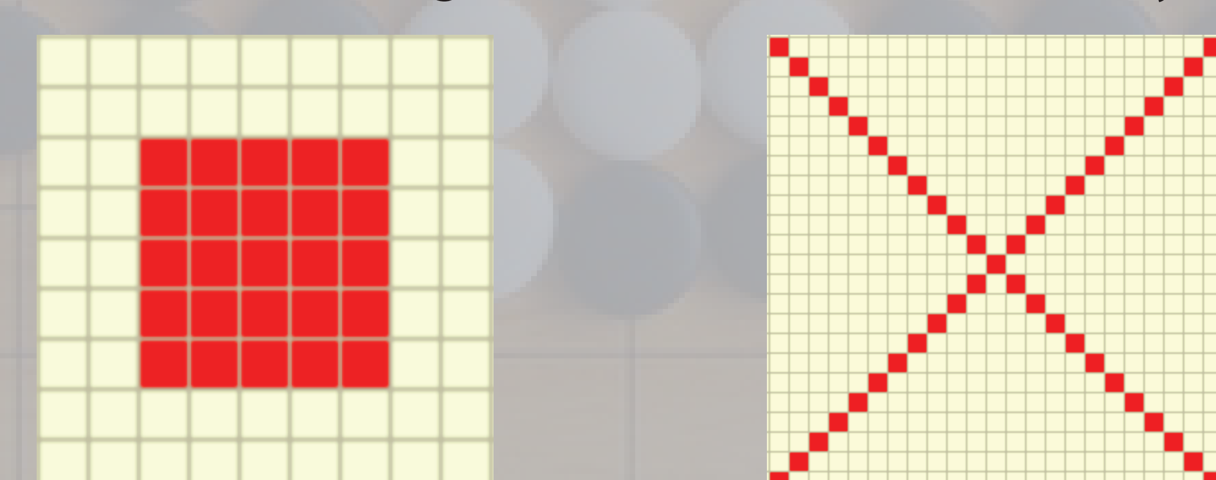


Figure 2:
Left: Vanilla 5x5 Square Convolution.
Right: 23x23 1-Width Diagonal Convolution.

Model

Our final model consisted of a cross layer (as seen in Figure 3), followed by 7 3x3 convolutions, then repeated. The final layer is a 1x1 convolution with 1 filter, giving scores for moves at each of the points on the 19x19 board.

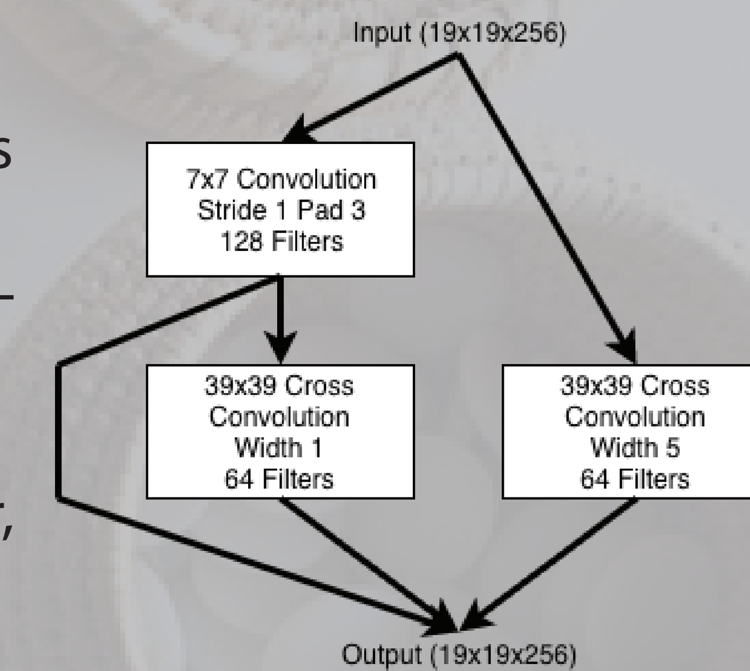


Figure 3: A cross layer.

Results

We evaluated our model on both playing strength in both in-person and online matches as well as classification accuracy on our test set of professional games. We were able to obtain a classification accuracy of 47%. The player was also able to beat intermediate amateur players.

Conclusion

We have successfully created a player that is capable of playing intuitively along. Amazingly, the player is also capable of understanding basic patterns, formations, and playing aggressively. Future work would involve expanding the model so it plays more effectively, understands the game at a deeper level, and knows when to pass or resign.