1 Summary of the article "Mastering the game of Go with deep neural networks and tree search"

The game of Go was known as one of the most difficult to solve problem due to its big search space and various positions and possible moves. In their work authors propose search algorithm based on Monte Carlo simulations combines with two methods "value networks" to evaluate position on the board and "policy networks" to choose move, which achieves 99.8% win rate.

There are in total $250^{2}50$ possible moves in Go. But the search space could be reduced with the following two approaches:

- Truncating the tree and replacing its value by the approximated evaluation function
- Sampling from probability distribution

Monte Carlo tree search applies Monte Carlo simulations to evaluate the value of each state in a search tree.

For go they apply convolutional neural networks that proved efficient for image recognition for board representation.

The first phase is to perform supervised training using human expert. Then next they train the reinforced learning policy network that improves the SL policy allowing to maximize winning in games. And lastly they train the a value network to correctly predict the winner in the game.

During the first stage the policy network switches between the convolutional layers with weights and rectifier nonlinearities. In the last layer we obtain the probability distribution over all possible moves. It is trained using the 30 million positions by applying the stochastic gradient ascent in order to maximize the likelihood of the human move selected in a particular state.

The second stage is trained by gradient reinforcement learning. The initial values are settled equal to those in SL policy. They randomize the pool of opponents in order to prevent the overfitting. The weights are updated at each time step by stochastic gradient ascent, which maximizes the expected outcome, which allowed to win in 85% cases against program called Pachi.

The final stage focuses on the evalution of the board position by estimating a value function, which is aimed at predicting the outcome of the game from the particular position of both players.

By lookahead search the alpha go select actions based on the policy and value networks in

MCTS algorithm. At each node the search tree stores the values of action, visit count and prior probability.