Marifel Barbasa
Udacity AISND March 2018 Cohort
June 2, 2018

Research Review on the AlphaGo Paper
Mastering the game of Go with deep neural networks and tree search
authored by researchers from the Google DeepMind team

The DeepMind researchers' goal was to tackle the challenging game of Go by creating an AI that would perform beyond amateur level play. The AI agent called AlphaGo eventually went on to defeat the human European Go champion Fan Hui. This is a groundbreaking feat because never before had a computer Go program defeated a human professional player, without handicap. AlphaGo has also been tested against other computer Go programs via an internal tournament, in which each program was allowed a computation time of 5 seconds per move; the results were that AlphaGo won 494 out of 495 games, giving it a 99.8% win rate. To construct an AI that would perform at the level of the strongest human players, the researchers implemented deep neural networks with Monte Carlo tree search (MCTS) and combined supervised learning (SL) and reinforcement learning (RL) techniques.

The AlphaGo neural network training pipeline consists of three stages: the SL policy network and the rollout policy, the RL policy network, and the value network. In the first stage, the SL policy network has 13 layers trained on 30 million board positions from the KGS Go Server and predicted human expert moves on a held-out test set with 57.0% accuracy. The rollout policy trained was faster but less accurate, achieving an accuracy of 24.2% with just 2 microseconds to select an action, as opposed to the 3 milliseconds for the policy network.

In the second stage of the training pipeline, the RL policy network improves upon the SL policy network. The RL policy network structure and weights are initialized to the SL policy network's, and then policy gradient reinforcement learning is applied to maximize the outcome of winning more games against previous versions of the RL policy network. This in turn generated a new dataset of games of self-play, consisting of 30 million distinct positions.

The final stage of the training pipeline makes use of this self-play dataset in a value network, which is trained by regression to predict the expected outcome (does the current player win?) given board positions. The value network has many convolutional layers but only outputs a scalar value. Comparing the self-play dataset with the KGS dataset, the degree of overfitting on the test set was reduced when using the self-play dataset on the value network.

Finally, the policy and value networks are combined with the MCTS algorithm to select actions via lookahead search. Tree traversal is done via simulation from the root state, and at each time step, an action is selected that would maximize the action value. After search is completed, the algorithm selects the most visited move from the root position. For AlphaGo to perform optimally using MCTS with deep neural networks (requiring orders of magnitude more computational power than conventional search heuristics), asynchronous multi-threaded search simulations are executed on CPUs, while policy and value network computations run in parallel on GPUs.

AlphaGo’s novel approach is to combine tree search with policy and value networks, working together harmoniously at scale. Previously, computer Go programs have only achieved weak amateur level play due to Go’s large breadth and depth of a search space. But now with AlphaGo’s demonstrated effectiveness in combining tree search with policy networks to select moves and value networks to evaluate board positions, thereby achieving human professional level play, there is greater hope that human-level performance can be achieved in other seemingly complex AI domains.