Timon Willi

Research Review

Summary

In march 2016 AlphaGo beat Lee Sedol, one of the very best Go players in the world, 4 to 1. “Mastering the game of Go with deep neural networks and tree search” explains in depth how the team at Google DeepMind were able to master a challenge that was thought to be unbeatable for another decade.

Previously existing Go-AIs are using Monte Carlo tree search (MCTS), refined with policy and value functions of linear nature based on human expert knowledge. With the renaissance of Deep Learning in the past few years, it can only be considered a logical step trying to replace those linear policy and value functions with their non-linear Deep Learning equivalents. Exactly this is what the team behind AlphaGo has achieved.

Before looking at how they designed their policy- and value-networks, it is necessary to understand the goal of said networks and MCTS in general.

Basically MCTS goes through all possible plays given enough time. It then chooses the play that gives the highest chance of winning, or in other words, it chooses the branch of the search-tree with the most variations that end up in a win. To improve MCTS, policies, based on human expertise are being used, such that MCTS can focus on more promising branches instead of wasting time on hopeless branches. The performance of the MCTS is ultimately highly dependent on the quality of those policies.

The contribution of the paper lies in the combination of the MCTS with deep learning, as well as the pipeline, which is being used to train the policy and value networks.

The product of the pipeline is a roll-out policy, a policy network trained with supervised learning (SL), a policy network trained with reinforcement learning (RI) and a value network. All of these 4 elements have different functions to fulfil the same purpose, reducing depth and breadth of the search tree: The rollout policy is a smaller, faster SL policy network. They try to predict expert moves based on the human-played games they have seen. When the SL policy network has finished training, it starts playing games against itself. It has then become the RI policy network. The goal of playing against itself is to finetune the SL policy network, such that it does not focus on predicting the correct move, but to win the game. Finally, the value network is trained on all the games played between the RI policy network and itself. Its goal is to predict the winner.

In the end, the algorithm works as follows: When traversing the tree, each node gets an probability value from the SL policy network called P(s,a), it gets an action value Q(s,a), which is a combination of the evaluations of the rollout policy and the value network. The algorithm then chooses the edge where the sum of those two values achieve its maximum. There are minor details in the equation to encourage discovery of new branches.

Key Results

The resulting algorithm introduces three new key components to the world. First there are the evaluation and move selection functions, second is the combination of supervised learning and reinforcement learning seen in the pipeline, where the RI policy network is based on the SL policy network and last but not least the combination of the lookahead search and the previously mentioned evaluation techniques.

AlphaGo has repeatedly beaten older Go-AIs based on older implementations of the MCTS with high winning rate and in the end, for the first time in history, a professional human Go-Player.