Research Review by Murat Senel

“Mastering the game of Go with deep neural networks and tree search”

Summary

This paper introduces the methodology by Google DeepMind, which enabled a computer program to defeat a human professional player in a full-sized game of Go. The authors introduce a new approach based on: 1) Value Networks and 2) Policy Networks. Value Networks are used to “evaluate board positions” and Policy Networks to “select moves”. The neural networks were trained by a combination of supervised learning and reinforcement learning from games of self-play. The authors also introduce a “new search algorithm that combines Monte Carlo simulation with value and policy networks”. This search algorithm, the AlphaGo program could win against other Go programs with a 99.8% success rate and could defeat the human European Go champion by 5-to-0.

Methodology

The approach consists of 1) training of the policy and value network and 2) efficiently combining these with Monte Carlo Tree Search (MCTS).
Hence the training pipeline consists various stages of machine learning:

- Supervised Learning (SL) Policy Network: Via stochastic gradient ascent, the policy network is trained to maximize the likelihood of human move in a given state. As a result, human expert moves can be predicted with an accuracy of 57%. The authors have also developed another version, which is less accurate (24% as opposed to 57%) but is much faster.
- Reinforcement Learning (RL) Policy Network: This stage improves SL-trained policy network by policy gradient reinforcement learning. The RL policy and SL policy network are identical in structure and their weights are initialized to the same values. By playing games between the current policy network and randomly selected previous policy network, this stage prevents overfitting. As a results RL policy network wins 80% of the games against the previous stage of SL Policy Network. Using no search at all, RL Policy Network wins 85% of games against Pachi, which according to the authors is the strongest open source Go program.
- Reinforcement Learning of Value Network: This stage estimates a value function to predict the outcome of position of games played with RL policy for both players. The authors have chosen to train the model with randomly sampled data-set, which they generated. The alternative of using data from complete games seem to lead to overfitting.

This three-stage pipeline is combined in an MCTS, which decides on actions by lookahead search. This stage starts with traversing the tree via simulation, starting from the root state. At each time of each simulation an action is selected from a given state based on an action value plus a bonus, which is function prior probability but decreases with repeated visits in order to stimulate exploration. These values are updated with the policies and value networks after reaching leaf nodes, in order to maximize the probability of selecting the best decisions. The algorithm chooses the most visited move from the root position.

Conclusion

The final version of AlphaGo used 40 search threads, 48 CPUs, and 8 GPU, and there is a distributed version running multiple machines, 40 search threads, 1,202 CPUs and 176 GPUs. The distributed version was tested against Fan Hui, the 2013,2014 and 2015 European Go champion and won the match by 5-0.