Research Review of

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

Summary:

The paper introduced a new approach that used "value networks" to evaluate the board positions and "policy networks" to select moves, deep neural networks trained by a novel combination of supervised learning and reinforcement learning. The paper also introduced a new search algorithm that combined Monte Carlo simulation with value and policy networks.

Techniques:

Recursively computing the optimal value function in a search tree could lead to a high winning rate of games. However, exhaustive tree search is infeasible for large games such as chess and Go. To reduce the complexity and improve the efficiency, there are 2 general principles:

- First, the depth of the search may be reduced by position evaluation. This has led to superhuman performance in chess, but not in Go due to the complexity of the game
- Second, the breadth of the search may be reduced by sampling actions. This can lead to superhuman performance in amateur level play in Go

The paper used neural networks to reduce the effective depth and breadth of the search tree: evaluating positions using a value network and sampling actions using a policy network.

To train those neural networks, both supervised learning and reinforcement learning were used. The training pipeline consisted of three stages:

1. <u>The first stage of the training pipeline: Supervised learning of policy networks</u>
   They used supervised learning (SL) to train policy networks directly from expert human moves.
2. <u>The second stage of the training pipeline: Reinforcement learning of policy networks</u>
   To improve the policy network, reinforcement learning (RL) was used. The RL policy network is identical in structure to the SL policy network. The RL policy network improved the SL policy network by optimizing the final outcome of games of self-play.
3. <u>The final stage of the training pipeline: Reinforcement learning of value networks</u>
   Use the RL policy network to estimate the value function.

AlphaGo combined the policy and value networks in an Monte Carlo tree search (MCTS) algorithm.

Results:

- AlphaGo evaluated thousands of times fewer than Deep Blue did, by selecting positions more intelligently using policy network, and evaluating them more precisely using the value network.
- By combining tree search with policy and value networks, AlphaGo has finally reached a professional level in Go – 99.8% winning rate against other G programs and defeated the human European Go champion by 5 games to 0.