Research Review

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

By Jiyun YANG

Summary

Because of its enormous search space and the difficulty of evaluating board positions and move, the game Go has been the most challenging game for AI. The paper introduces a new approach to computer the game Go using "value networks" to evaluate board positions and "policy networks" to select moves. It has implemented deep neural networks by combining supervised learning from human expert gams, and reinforcement learning from games of self-play. In addition, the paper also introduces a new search algorithm that combines Monte Carlo simulation with value and policy networks. The implementation of this paper "AlphaGo" achieved a 99.9% winning rate against other Go programs and defeated the human European Go champion by 5 to 0.

Techniques introduced

1. Supervised learning of policy networks (SL policy network)
   This paper builds a 13-layer CNN trained on 30 million Go game positions. By taking the board state as input, it alternates between convolutional layers with weights and ReLU activation functions. The final softmax layer outputs a probability distribution over all legal moves.
2. Reinforcement learning of policy networks (RL policy network)
   RL policy network uses an identical structure and initial weights as the SL policy network. The game is played between the current policy network and a randomly selected previous iteration of the policy network. It won more than 80% of games against the SL policy network.
3. Reinforcement learning of value networks
   This step focuses on position evaluation. It estimates a value that predicts the outcome from a position of games played by using policy for both players. It outputs a single prediction instead of a probability distribution.
4. Searching with policy and value networks
   AlphaGo combines the policy and value networks in an MCTS algorithm that selects actions by lookahead search. The SL policy network performed better than the stronger RL policy network whereas the value function derived from the RL policy network performed better than a value function derived from the SL policy network. Also, to efficiently combine MCTS with deep neural networks, an asynchronous multi-threaded search has been used, which executes simulations on CPUs, and computes policy and value networks in parallel on GPUs.

Results

This paper implemented two versions of AlphaGo (single-machine and one distributed), both of them significantly outperforms any previous Go program. The distributed version defeated the European Go champion by 5 to 0.