Game of Go summary

-xiao, wenhao

For large games with perfect information such as Go and chess, two general principles are drawn to solve the NP problems involved. The first principle is to reduce the depth of the search, which can be done by using position evaluation. The other principle is to reduce the breath of the search, which can be done by using sampling actions like MCTS. However, prior works have been limited to shallow policies or value functions based on a linear combination of input features.

For Alpha Go, the position evaluation and sampling actions are implemented by using a similar architecture to Convolutional Neural Network, and the architecture pipeline is as follows. First, it trains a Supervised Learning Policy Network with input data from expert human moves to get high quality gradients. Then, it trains a fast policy that can rapidly sample actions during rollouts. After that, it trains a Reinforcement Learning Network to improve Supervised Learning Network. Lastly, it trains a Value Network to predicts the winner of games by self-competing.

Let’s first look at the Supervised Learning Network. The input data of the neural network comes from 30 million positions from the KGS Go Server, which gives a simple representation of the board state. Given the board state ,the trained neural network will give the prediction of the next move of the human expert. The neural network achieved 57.0% accuracy compared with 44.4% accuracy in previous model. Small improvements in accuracy led to large improvements in playing strength. For even better performance, The pipeline uses Reinforcement learning to improve the previous Supervised learning.

Now, let’s focus on position valuation. in this stage, it trained a Reinforcement Learning of Value Networks to approximate the value function. The RL network architecture for value function is similar to previous RL network, but gives only one output instead of a probability distribution .

AlphaGo Combines the policy and value networks in an MCTS algorithm that selects actions by lookahead search.The SL policy network performed better in AlphaGo than the stronger RL policy network, presumably because humans select a diverse beam of promising moves, whereas RL optimizes for the single best move. However, the value function derived from the stronger RL policy network performed better in AlphaGo than a value function derived from the SL policy network.

Lastly, To evaluate the playing strength of AlphaGo, testers run the tournament against other commercial Go programs, the results of the tournament suggest that single machine AlphaGo is many dan ranks stronger than any previous Go program, winning 494 out of 495 games (99.8%) against other Go programs, and AlphaGo won the match 5 games to 0 against human professional player Fan,hui.

In this work google engineers have developed a Go program, based on a combination of deep neural networks and tree search, that plays at the level of the strongest human players. They have developed, for the first time, effective move selection and position evaluation functions for Go, based on deep neural networks that are trained by a novel combination of supervised and reinforcement learning. They also introduced a new search algorithm that successfully combines neural network evaluations with Monte-Carlo rollouts. Their program AlphaGo integrates these components together, at scale, in a high-performance tree search engine.