Vasiliy Mikhailov
Udacity Artificial Intelligence Nanodegree
9 March 2017

Alpha Go

Research review

Review is based upon Mastering the game of Go with deep neural networks and tree search article https://storage.googleapis.com/deepmind-media/alphago/AlphaGoNaturePaper.pdf

The game of Go has long been viewed as the most challenging of classic games for artificial intelligence owing go its enormous search space and difficulty of evaluating board positions and moves. Game of Go has breadth (number of legal moves per position) of 250 and depth (game length) of 150, which gives search field of 250 ^ 150. AlphaGo achieved 99.8% winning rate against other Go programs, and defeated the human European Go champion by 5 games to 0.

AlphaGo uses deep neural networks to make it's moves in two different ways: 'value networks' are used to evaluate board positions and 'policy networks' are used to select moves.

Those networks are trained by combination of supervised learning from human expert games and reinforcement learning from games of self-play.

Learning pipelined as following: first goes supervised learning, trained on 30 million positions from KGS Go Server, then goes reinforcement learning, where current policy network plays game with randomly selected previous iteration of the policy network and then reinforcement learning of value network is performed.

Policy network 13 layers, trained with supervised learning (SL network), it predicted expert moves with an accuracy of 57,0% using all input features and 55,7% using only raw board position and move history as inputs, compared to 44,4% from other research groups.

Reinforcement learning (RL) policy network won more than 80% of games against SL-policy network.

Value network has similar architecture to policy network, but has only one output instead of probability distribution. Single evaluation of value network approached the accuracy of Monte Carlo rollouts using RL policy network, but required 15,000 less computation.

AlphaGo combines the policy and value networks in an Monte Carlo tree search algorithm, that selects actions by lookahead search.

To efficiently combine MCTS with deep neural network, AlphaGo uses an asynchronous multi-threaded search that executed simulations on CPUs and computes policy and value networks in parallel on GPUs. The final version of AlphaGo used 40 search threads, 48 CPUs and 8 GPUs. Distributed version of AlphaGo was implemented, that exploited multiple machines, 40 search threads, 1202 CPUs and 176 GPUs.

To evaluate strength of AlphaGo, tournament between AlphaGo and best current programs/algorithms was conducted. Each move was limited to 5s. AlphaGo won 494 of 495 games (99,8%). Distributed version of AphaGo won 77% over single machine AlphaGo. It was proven, that even turning MCTS in AplhaGo off, it beats other MCTS programs, demonstrating that value networks provide a viable alternative to Monte Carlo evaluation in Go. Finally, distributed version of AlphaGo was evaluated against Fan Hui, winner of 2013, 2014, 2015 European Go championship. This was the first time that a computer Go program has defeated a human professional player.

During match against Fan Hui, AlphaGo evaluated thousands of times fewer positions than Deep Blue did in its chess match against Kasparov; compensating by selecting those positions move intelligently, using the policy network and evaluating them more precisely, using the value network. While Deep Blue relied on handcrafted evaluation function, the neural networks of AlphaGo are trained directly from playing.