research_review

May 12, 2017

The game of Go was considered as the most challenging among classic games due to its enormous search space. The DeepMind Team was aiming to create program that achieves superhuman performance in Go. Their program AlphaGo was built by wrapping up two elements, policy network and value network into Monte Carlos tree search (MCTS), which is also the common foundation for the strongest Go programs ever created before AlphaGo (Crazy Stone, Zen, Pachi and Fuego).

Each round of MCTS consists of four steps: selection, expansion, simulation and backpropagation. The model starts selection by using stored prior probability for each node given state. By randomly rolling out the game, the next three steps updated probability for each node and thus help the game tree expand towards the most promising moves.

Policy network was used to narrow search space while value network was used to quickly evaluate action given certain state. For different purposes, they trained three different policy network: a fast but less accurate network trained by a linear softmax of small pattern features P_1; a slow but more accurate network trained by stochastic gradient ascent P_2; an improved network by policy gradient reinforcement learning P_3. To estimate value network and avoid over-fitting, they generated a new self-play data set consisting of 30 million distinct positions by playing games between the P_3 and itself and trained value network on this data set using stochastic gradient descent to minimize the mean square error between predicted value and corresponding outcome. In the expansion step of MCTS, they updated stored probability for each action by P_2. In the evaluation step, they evaluated node (action- state combo) in two different ways:

1. they run a rollout to the end of the game by P1 and evaluate outcome;
2. they evaluated the node by using value network directly.

The final model have these two evaluations combined using a mixing parameter.

They evaluated the performance of AlphaGo by running an internal tournament among variants of AlphaGo and several other Go programs. The results of the tournament showed that the single-machine AlphaGo was much stronger than any previous Go program, winning 99.8% of game against other Go programs in the tournament and the multiple-machine AlphaGo was significantly stronger than single-machine version. More importantly, distributed version of AlphaGo won 5-0 against a professional 2 dan, and the winner of the 2013, 2014 and 2015 European Go championships in a formal five-game match. This is the first time that a computer Go program has defeated a human professional player without handicap in the full game of Go.