Research Review

Chenguang Yang

This is a brief review on the goal, techniques and results of AlphaGo research, which was published on Nature in Jan-2016 as "Mastering the game of Go with deep neural networks and tree search". This review mainly serves readers who have entry-level knowledge in AI game playing.

Goal

The AlphaGo research team aims to develop an AI agent that can play Go at human master's level, and this goal came up with awareness of both technical challenge and current progress among peers.

The game of Go has been widely viewed as the most challenging classic games in the AI field, due to its enormous number of state of a game and difficulty to quantify a position or move into utility functions. Before AlphaGo, the strongest Go programs were Crazy Stone (commercial) and Pachi (open-source). Both have arrived at the strong end of amateur player, and AlphaGo targets one level higher, which is professional tier in KGS benchmark.

Techniques

First comes my understanding about the overall structure of several techniques used in AlphaGo, followed by more details on each component, and discussion on what was newly introduced.

Similar to human Go player, AlphaGo gets some **training** before **playing** a real match with time constraint. AlphaGo applies **Monte Carlo Tree Search** (MCTS) in live games, combined with 2 **Convolutional Neural Network** trained beforehand, one for policy and the other for value.

1. Monte Carlo Tree Search

Instead of traversing all possible branches at each state of the Go game tree, this search strategy runs Monte Carlo simulation to estimate the value of tree nodes, and this kind of simulation is based on a set of **policy**, in plain words, what moves to expect. Fundamentally speaking, MCTS reduces the breadth of a search tree to a feasible situation. However, the depth of a Go game tree would make MCTS still ineffective if there were no **value** evaluation, also known as heuristic/utility function.

MCTS doesn't originate from this research, and has been used in previous Go agents. AlphaGo combined MCTS with some new techniques in policy and value, which were very simple representations in achievements that preceded AlphaGo.

2. Policy Network

There are 3 different policy developed, with different accuracy-time tradeoff. The least accurate but fastest one uses limited board pattern features, and simpler algorithm. The next **SL policy** is built with supervised learning, and behaves more like human Go expert. Last one is **RL policy** built with reinforcement learning, which improves SL policy through tons of self-play. Certainly, RL is most time consuming, but has the strongest neural network, because it adjusts SL towards actual winning.

The first 2 policies are from prior research work, and RL policy is a new contribution by AlphaGo. With 3 policy candidate, next stage is to train a value network. Note that value network is dependent on policy network when being trained, although it can be used alone in live gameplay.

3. Value Network

This last step is a neural network based regression model that predicts the outcome of given game state, assuming game is played by certain policy of both sides. After experiments, AlphaGo team found that best value network is trained by RL policy. Additional attention was given to overfitting, and the solution is to generate a new dataset using self-play and avoid the overfitting caused by highly correlated successive board states.

Value network is a new advancement, in contrast against prior Go agents, which had linear combination of input features as value estimation.

Results

The performance of AlphaGo has been showcased by internal tournament against other Go programs, and also 5 games against European Go champion, Fan Hui. AlphaGo ran on a single machine with asynchronous multi-thread and multiple CPUs and GPUs in the internal tournament, and had a dominant 99.8% winning rate against other Go agents. And AlphaGo was upgraded to a distributed version in the series against human Go champion, and swept a 5-0 victory.

It hadn't occurred yet by the time of this research paper, but in Mar-2016 AlphaGo beat Lee Sedol 4-1, one of the strongest human player in the history of Go. This became a solid testimony of the amazing result of AlphaGo research, and also a big milestone in AI history.