Research Review on: Mastering the game of Go with deep neural networks and tree search

by Luka Anicin

Friendly introduction to a game of GO[1]

A game of Go starts with an empty board. Each player has an effectively unlimited supply of pieces (called **stones**), one taking the black stones, the other taking white. The main object of the game is to use your stones to form territories by surrounding vacant areas of the board. It is also possible to capture your opponent's stones by completely surrounding them. The normal size board for the GO is 19 x 19!

Even though the rules of the game are not so hard to learn, mastering this game can be really challenging for even a human being. The master players of this game are playing and practicing it for decades to be what we call professional player.

Why Go is hard for AI?[2]

There are two main reasons.

1. Search space is much larger to any other game (example chess) that computers took a chance to conquer. To put this in perspective, an average numbers of moves per turn in Go is about 200 (two hundred) comparing to chess where we average number of moves per turn is 35.
2. Heuristics for Go are not so clear. The object of the game is to capture the territory, but there is not an easy way to evaluate if some territory is captured or not, before the board is complete.

DeepMind Goals

The main goal for DeepMind was to master the game of Go using deep networks and tree search, which is as well mentioned in the title of the paper.

As we know game tree for Go is very big and exhaustive search is not an option. By regular formula it is $\mathbf{b}^{\mathbf{d}}$ where **b** is breadth of the game tree and **d** is depth of the tree. In the case of Go it would be approximately 250^{150} , it is not possible to perform classical search and determine the best move.

Before DeepMind made AlphaGo, it was thought that Go will be unconquered by a computer for decades in the future.

By solving this game DeepMind has proven that Artificial Intelligence is advancing very fast and it can and will be used to solve many other problems in everyday life as well.

AlphaGo Results

In 2015. AlphaGo successfully defeated human player in Go, without handicap, 5 to 0.

Over internal testing DeepMind has tested AlphaGo on 495 games against other Go programs and AlphaGo won 494 games which is 99.8% win rate!

---

[1] https://www.britgo.org/intro/intro2.html - British GO association

[2] https://www.youtube.com/watch?v=LEtVi9qTY84 - Udacity YouTube, Why Go Is So Difficult For AI...

In this article DeepMind reviled what it took, from hardware side, to evaluate this model.

"*The final version of AlphaGo used 40 search threads, 48 CPUs, and 8 GPUs. We also implemented a distributed version of AlphaGo that exploited multiple machines, 40 search threads, 1,202 CPUs and 176 GPUs."*

Techniques they've used

- For a search algorithm they implemented Monte Carlo tree search (MCTS) which is combined with 'value' and 'policy' networks.
- To reduce search time they've used **value networks** to evaluate positions on the game board and **policy networks** to select moves.

The interesting thing about their **training pipeline** is that they have used **supervised learning** and **reinforcement learning** approach combined.

References

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