Summary: AlphaGo by DeepMind Team

By Simon Müller

Goals and Applied Techniques

The goal of this paper is to describe a technique for mastering the game GO by artificial intelligence, more explicit by deep neural networks, reinforcement learning, and tree search. The program combining these techniques was named as AlphaGo.

More detailed, the AlphaGo team needed to solve games with massive search space by applying novel techniques. The computer has to evaluate in for each board position or state a value function that needs to search each possible own and opponent move from that state (GO is a full information game). Therefore, the player or computer must search approximately b^d of sequences of moves in such kinds of games (where b is the game's breadth and d is the depth). Compared to chess, the search space for GO games is much more massive. In numbers: for chess, we have $b \approx 35$ and $d \approx 80$ and for GO $b \approx 250$ and $d \approx 150$. This complexity of GO was a long time a big hurdle and unconquerable for computer programs.

The AlphaGo team captured this problem and solved it. Furthermore, there aim is not just solving board games, they even want to use their gained experience during this challenge to solve real world problems.

In summary, the AlphaGo team applied several novel techniques from machine learning and artificial intelligence in this challenge:

- They used *value networks* for evaluating board positions and *policy networks* for selecting moves.
- Human expert games were used for training deep neural networks combined with supervised learning. Furthermore, reinforcement learning was applied in self-playing games.
- AlphaGo developed a new search algorithm that combines Monte-Carlo simulation with the value and policy networks mentioned in the first point.

Results

The first time in history and unexpected by public, a computer (AlphaGo) defeated high rated and professional human Go players. The last frontier for this kind of game was conquered by artificial intelligence.

Before AlphaGo played against humans, the AlphaGo team benchmarked their software against other computer GO software. The result was that AlphaGo is many *dan* ranks stronger than any other Go software, more precisely AlphaGo had overall a winning rate of 99.8%. Even with a handicap, AlphaGo is superior to other GO software. Furthermore, the AlphaGo team evaluated the computational needs for such a task. They developed a single pc (multithreaded and GPU based) and a distributed version of AlphaGo. The difference in the Elo value from the single version to the distributed version was relatively low (2,890 to 3,140) compared to the best competitor software (1,929). This emphasizes the superiority of the applied method compared to the need of compute power during the game.

As already mentioned earlier in this summary, the AlphaGo team work awake the hope that human-level performance in other artificial intelligence domains can be achieved with novel techniques (or by applying this technique) which were previously seen as unsolvable.