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

The research paper describes approach to develop a program *AlphaGo* that can beat a human professional player in a game of Go, which is a classic board game in which two players compete to surround more territory on a board of 19 x 19 grid lines. Development of a successful computer player was a big challenge for programmers due to the complexity of the game which can be evaluated by number of sequences of moves, which for Go is around 250^{150} whereas for chess it is only 35^{80}, the number significantly lower.

AlphaGo uses powerful evaluation functions are not handcrafted (as it was in case of winning *DeepBlue* chess computer player), but based on neural networks trained directly from the game, which are part of *its three components to* evaluate board position and moves. Two first components are neural networks that are trained by novel combination of supervised and reinforcement learning i.e.:

- Policy Network, which is trained on high-level games to imitate best Go human players. Policy Network first scans the positions selects spots to play, which are evaluated more precisely by the second component, Value Network.
- Value Network takes the outcome of Policy Network for both players and evaluates the board position and probability of winning in this particular position.

Those neural networks reduce effective depth and breadth of a search tree; therefore the number of sequences of moves is significantly smaller.
The third element, Tree Search looks into different variations of the game and tries to figure out what will happen in the future (lookahead search). It combines neural network evaluations with Monte Carlo rollouts (MCTS – Monte Carlo Tree Search). All available similar software is based on high – performance MCTS algorithms. *AlphaGo* tries to maximize its probability of winning, but it doesn't care about the margin by which it wins, which is a different approach to a human brain that often falls into a trap of maximizing the score which is not required to win, as long as the winning player has an advantage of one point.

AlphaGo was tested against other computer players and human professional player. When it competed against other Go software *AlphaGo* achieved very high winning rate of 99.8%, which proved that neural networks greatly improved Go evaluation mechanisms in comparison to MCTS without neural network support.
The second test against human professional player also was successful. *AlphaGo* defeated human professional champion (Fan Hui) for the first time in the history of Go computer players (score 5 to 0). This success wasn't expected for at least another decade.

The success of AlphaGo opens door to solving other complex challenges, in which human brain still wins with AI algorithms. By applying neural networks in tree search algorithms, AI might finally outperform human capabilities in other elaborate, decision-based tasks.