Research Review: AlphaGo

Sagarnil Das

10/30/2017

Google Deepmind's latest miracle in the field of AI is the birth of AlphaGo, an AI agent to defeat the world champion Go player. Go and Chess, these two games are noted to be two of the most complicated games in the history. Chess has an average branching factor of 35 and an approximate depth of 80. Go has an average branching factor of 250 with an approximate depth of 150. So naturally, due to the vast search space and game tree size, traditional game playing methods like minimax and alpha beta pruning will prove to be ineffective. On top of that, the strategies of Go is much harder to express in algorithmic terms, thereafter making it equally difficult to instill human knowledge in an AI agent.

In recent years, before AlphaGo, strides were made in the domain of Go with the help of Monte Carlo Tree Search(MCTS). MCTS uses Monte Carlo rollouts to estimate the value of each state in a game tree. As more simulations are executed, the search tree grows larger and the relevant values become more accurate. This establishes a decent probability of winning for the possible moves which are simulated, and is known to slowly converge upon the same decision tree as minimax even though it doesn't have a scoring function. However, in real world, this will be still too large a sample size and hence not feasible to play the game and complete it in time as the game tree still would be huge.

This is where the novelty of AlphaGo lies. Even after implementing MCTS, it still needs a way to reduce the possible moves to consider in order to deeply search a much fewer moves of much higher value. It utilizes two Deep Neural Networks (DNN), known as the policy network and the value network. In the policy network, a supervised learning (SL) policy network is trained directly from expert human moves. This provides fast, efficient learning updates with immediate feedback and high-quality gradients. Next, it is trained by a reinforcement learning (RL) policy network that improves the SL policy network by optimizing the final outcome of games of self-play. Finally, a value network is trained that predicts the winner of games played by the RL policy network against itself. So the policy network helps focus the MCTS on paths that are of high likelihood of actually occurring.

With the effective combination of MCTS and the two DNNs, AlphaGo was able to achieve a superhuman performance level outperforming every other computer Go playing agent. Then, in a Go series, it defeated Fan Hui, a highly ranked player and a world champion of Go. Recently, the move analysis of the playing method of AlphaGo has revealed that it has developed unknown strategies which are not part of the professional theory and understanding of Go. This is indeed a revolutionary achievement which is also a bit scary!