Christopher Verceles

AlphaGo Research Review

Beating expert human players at the game of Go has long been a difficult challenge for intelligent agents. Several reasons, such as the search space being too large to traverse through brute force alone, kept AI from advancing past this milestone until recently. DeepMind's AlphaGo agent was, for the first time, able to improve upon previous techniques, such as the Monte Carlo Tree Search (MCTS) and typical evaluation heuristics, to effectively combine "intuition" and reflection in choosing possible winning moves and as a result play more like human experts. These improved techniques were achieved using neural networks, which are basically collections of interconnected "nodes" designed to function like a basic human brain, thus the term "neural network".

These value and policy neural networks were trained using supervised learning based on human expert games and games of self-play. Simply explained, the policy neural network actually consists of 2 major parts; the first is used as a sort of initial move set trained from expert human players, while the second is trained to optimize the final outcome resulting from the first policy neural network. The value neural network , on the other hand, is trained to predict the winner of a game based on training data from playing against itself. Finally, these several layers of neural networks are used to evaluate a certain state with MCTS to create an expert level intelligent agent that can and has won against several of the worlds' best Go players.

AlphaGo's novel algorithms, used in tandem, achieved significant improvements over previous Go agents based purely on MCTS and basic evaluation functions that only achieved amateur player levels. The policy neural networks effectively replicated "intuition", which drastically lessened the search space, thus reducing the needed computational resources, but still implementing reflection with the MCTS algorithm that machines can do faster than human players. AlphaGo used this implementation to beat the European Go champion by 5 - 0 and Lee Sedol, the second best Go player in the world, by 4 - 1. This AlphaGo implementation, however, is far from perfect. Lee Sedol, who simply learned from manual games one at a time was still able to learn on the spot and beat AlphaGo after a few games. AlphaGo, in contrast, needed 176 GPUs, 1202 CPUs (Burger, 2016) and a large dataset of human expert games and its own self play data sets to beat Le Sedol. Though limited, this achievement brings AI agents much closer to achieving consistent superhuman level performance in other areas, such as image recognition, and problems leading to general purpose learning in search spaces that are impractical to completely traverse.

References:

Silver, Huang, Maddison., et al. (2016, January 28). Mastering the game of Go with deep neural networks and tree search. Retrieved from https://storage.googleapis.com/deepmind-media/alphago/AlphaGoNaturePaper.pdf

Burger, C. (2016, March 16). Google DeepMind's AlphaGo: How it works. Retrieved from https://www.tastehit.com/blog/google-deepmind-alphago-how-it-works/