Nikolas Katigbak
March AIND — Term 1
Project 1 — Advanced Game Playing Agent
Research Review

AlphaGo Summary

How did AlphaGo learn to play Go so well? Over the years the Monte Carlo Tree Search (MCTS) was developed as a method to better approximate a Game Tree and traverse it not just quicker but in a more efficient manner as well; picking branches of play that are statistically effective. AlphaGo took this concept to a whole new level and paired it with modern deep learning techniques.

Mainly there are two main types of networks that AlphaGo uses i) policy network and ii) value network. One of the policy networks is a supervised-learning network that learns Monte Carlo policies from a labelled data set of about 30 million games. The other policy network is a policy gradient reinforcement learning network that attempts to perfect the supervised-learning network's policy; it trains by playing against random previous versions of the policy network. The last policy network is a faster (less accurate) rollout policy that is used in conjunction with the first policy network in order to perform the MCTS. Thus for the policy networks of AlphaGo there is supervised learning of MCTS policies and also reinforcement learning (which takes the learned policies) by playing against random other/previous versions of the policies.

The other technique utilized by AlphaGo is the value network which is a reinforcement learning network that learns from, not boards configurations from the same game, but rather board configurations that each were from one in the 30 million games played by the policy network's self-play. This allowed the value network to not over-fit and fail to generalize; had it learned from board configurations from the same game there would be high correlation/s among them. This value network allowed AlphaGo to assess the value of each game-board configuration in terms of the likelihood of a win.

The policy network(s) are used in conjunction with the value network when performing a MCTS. Using the policy network to pick statistically effective branches as AlphaGo traverses down the tree. When a leaf node is encountered, in addition to the other faster policy network for rollout the supervised-learning, the value network is also used to evaluate the outcome. Surprisingly, even if the faster policy network is silenced and only the value network is used AlphaGo was able to still defeat other AI Go agents.

With that novel architecture AlphaGo was able to defeat European Champion Fan Hui and World Champion Lee Sedol. In addition, it was also able to successfully defeat other previous-reining Go AI agents: Crazy Stone and Zen (commercial) and Pachi and Fuego (open source). The single-host implementation of AlphaGo has 40 search threads, 48 CPUs and 8 GPUs while a multi-host (distributed) implementation of AlphaGo had 40 search threads, 1,202 CPUs and 176 GPUs; the distributed implementation of AlphaGo beat other Go AI agents 100% of games they played.