Research review of AlphaGo paper by DeepMind team

Overview: goals and techniques used:

The paper explores a new approach for using AI to create an agent for playing the game of Go. The approach combines “value networks” (evaluation function) and “policy networks” (selecting moves) in a Monte Carlo tree search (MCTS), where the MCTS searches the tree to maximum depth by random sampling. This is needed since exhausting searching the entire search tree in Go is intractable, having a search space of 250^{150} (breadth ~ 250, depth ~ 150).

For each board position, a 19x19 image (corresponding to the Go board dimensions) is fed into convolutional neural network layers to create a representation of the position that reduces the effective depth and breadth of the search tree. Supervised learning (SL)

These neural nets are trained by using a supervised learning (SL) policy network fed by expert human moves. A reinforcement learning policy network is used to improve the SL policy network by optimizing the final outcome. A value network is trained by regression to predict the expected outcome in positions from the self-play data set.

Results summary:

The charts below show the results after running a tournament test where AlphaGo was pitted against commercial and open-source Go programs as well as a world-class human Go player, and reveal AlphaGo to clearly perform even the human player. The best performance is achieved in the distributed case, where AlphaGo is run on many compute hosts in a cluster arrangement, versus on a single machine