AlphaGo: the first AI to achieve master in Go

A goal of the paper was to present the techniques that were used to create a Go playing AI that was capable of beating a human master level Go players. Go is similar to chess in a way that both are two player board games with each player taking turns and no element of randomness is present during the games. However, differently from chess, Go has a much larger number of possible game states and more actions available in each state. The combination of these factors made the Go playing AI impractical with the previously known methods. AlphaGo came up with several novel approaches described in the paper that allowed it to reach a master human playing level.

As we have learned in the advanced search topics presented in the course, games like Go and Chess involve search techniques that build game state trees and then search for the best moves by using various traversing optimization techniques. Since evaluating a full state tree is impractical, the game tree is iterated up to a certain level and then the state is evaluated with a heuristic that is usually built over time via research and game analysis that can accurately interpret the usefulness of the board for a player.

Where AlphaGo differs from other Go AIs is in a method it used to develop a system for game state evaluation and effective move selection. It combined a learned value and policy networks with Monte Carlo tree search (MCTS) rollouts for a very effective search algorithm to traverse the game state tree.

The effectiveness of the networks came from it first being built using supervised learning by using human expert Go games followed with reinforced learning from self-play. Instead of learning to predict expert human moves like previous AIs did, it learned the moves it should use from the self-play. Reinforcement-learned value networks strengthened the network produced from supervised learning which led to its dominance against existing Go AIs and eventually a human expert players.

Evaluating policy and value networks was very computationally intensive operation, much more so than compared to a more traditional search heuristics. To accomplish this in a reasonable time, policy and value networks are evaluated on GPUs while in parallel MCTS simulations are run on the CPUs. The AI also could be run in a distributed configuration across multiple machines where it achieves even more impressive win rates and Elo ratings.

Event without using any search and relying on just reinforced learned networks, AlphaGo could defeat the most sophisticated Go AI 85% of the time. The creators of the AI measured its effectiveness using just Monte Carlo tree search, just the policy network, just the evaluation network, and then the combinations of the three. The most effective results came from all three being used in playing the game where it defeated all the existing AIs by a wide margin (99.8% winning rate).

AI achieved the highest Elo rating from all the AIs as well as higher rating than a human player, European Go champion Fan Hui. Eventually it led to a highly publicized and 4-1 win against Lee Sedol. An imaginable feat for AI that was thought to be years from being able to compete with master human players.