AlphaGo Research Review

AlphaGo is a very famous computer program to play the game Go. In 2016, it even won the world champion Lee Sedol. Most recently, it even got a 60-win in an INTERNET gaming platform which is based in China. In this report, I will give a summary about the algorithm and techniques used by AlphaGo.

AlphaGo first trains two policy networks. One is a supervised learning (SL) policy network p_σ, which is used for predicting expert human moves. To build this policy network, they utilize convolutional neural networks (CNN). They describe the state of a board as a 19×19 image and use convolutional layers to construct a representation of the position. The CNN they develop has 13 layers and can predict the next position based on the position records from KDS. The second policy network is a fast rollout strategy, which can be represented as p_π and be used to evaluate the searching leaf node.

Further, they enhance the SL policy network by training a reinforcement learning (RL) policy network p_ρ. By self-play, RL policy network is able to optimize the final outcome of games. In addition to that, RL policy network also help training a value network v_θ.

AlphaGo deploy Monte Carlo Tree search to implement the search part. At the beginning of each simulation, the algorithm traverses the tree by selecting the edge with maximum action value (adding other items such as visit counts). When it gets the the node, it will expanded the node and the leaf position is just once processed by the SL policy network. The leaf node is evaluated in two different ways: by the value network $v_\theta(SL)$ and the outcome z_L of a random rollout played out until terminal step using the fast rollout policy p_π. These two values will be combined using a mixing parameter λ. At the end of simulation, they update all the action values and visit counts of all traversed edges. Finally, the position with most visited move from the root position is selected as the next position.

AlphaGo uses an asynchronous multi-threaded search on CPU and computes policy network and value network on GPUs. The final version contains 48 CPUs and 8 GPUs. They also has a distributed version of AlphaGo with more computing units.

At first, they evaluate the performance of AlphaGo by playing with other Go computer programs. The result is pretty good and the winning rates over other AI is above 70%. They further invited a professional player and finally won.

One thing should be noted is that, AlphaGo doesn't rely on a handcrafted evaluation function. All the policy network is just learned from the KDS data and self-play. AlphaGo, already becomes an great exemplary that researchers can use deep neuron network to solve hard problems.

References

[1] D. Silver et. al. 1991. *Mastering the game of Go with deep neural networks and tree search.* Nature vol 529.