AIND-Build a Game-Playing Agent

Research Review

Cheng Wang

Paper

Silver, D. *et al.* Mastering the game of Go with deep neural networks and tree search. *Nature* **529**, 484-489 (2016).

Summary

Go is considered as one of artificial intelligence's "grand challenges" due to its extremely high search space with approximate 250^{150} possible moves (breath is about 250 and depth is about 150). The brute-force search is infeasible for this challenge. In selected paper, the authors combine two state-of-art techniques (i.e. Monte Carlo Tree search (MCTS) and deep convolutional neural network) in computer Go playing and they have developed a program AlphaGo that could play at the strongest human player level.

In order to select move and evaluate position, policy network and value network are trained using deep convolutional neural network. Policy network outputs a probability distribution of legal moves given current board state for future move selection. Two types of policy networks are trained. One is a 13-layer supervised learning (SL) policy network p_σ, which is trained from 30 millions human expert moves. In addition, a rollout policy p_π is also trained on the same dataset for quick action sampling during rollouts. Another one is reinforcement learning (RL) policy network p_ρ, which is improved from SL policy network by policy gradient reinforcement learning. The weight ρ is updated by stochastic gradient ascent in the direction to maximize the expected outcome. Value network, which outputs the expected outcome for a possible position, are trained for position evaluation. A new dataset is generated from game self-play with the RL policy network and is used for value network training by regression with minimal overfitting.

AlphaGo uses Monte Carlo Tree search algorithm with policy and value networks to select the move in game playing. Each edge (s,a) in a search tree is associated with an action value $Q(s,a)$, visit count $N(s,a)$, and prior probability $P(s,a)$. The tree is traversed by simulation and the action with highest $Q(s,a) + P(s,a)/(1+N(s,a))$ value is selected for further exploring in each simulation. SL policy network value for action a given state s is store as $P(s,a)$ in tree searching. The value of leaf node is evaluated by both value network and outcome of the game by using fast rollout policy p_π. The results of tournament indicate that equal contribution of value network and rollouts performed best. The mean value of the leaf node of all simulations is used as active value $Q(s,a)$. The most visited move in the simulation from the root is selected.

After combining all the strategies together, AlphoGo can play at the level of the strongest human players. Single machine AlphaGo wins 494 out of 495 games against other Go programs such as Crazy Stone and Zen, while distributed version of AlphaGo wins 100% games against other programs. In addition, the distributed AlphaGo defeated the human European Go champion by 5 to 0.