Mastering the game of Go with deep neural networks and tree search

Go is considered one of the most complicated classical games for Artificial intelligent with 250^{150} possible move sequences, large then chess which has only 35^{80} possible move sequences.

The aim of the pepper is to introduce method which reduce depth and breadth of search using deep and reinforcement learning, to create agent which can out perform state of the art Go program which use Monte Carlo tree search.

The authors reduced depth and breadth of search tree by using convolutional neural network to analyse 19 x 19 board image to construct representation of the positions, and two neutral network. Value network to evaluate the positions and policy network to evaluate sampling actions.

The authors trained the neural networks using three stage pipeline, which consist of the following stages of machine learning:

- The first stage of the pipeline use supervised learning to train 13 layers policy network (SL) on randomly sampled pair of state and action, using stochastic gradient ascent to maximise the likelihood of the human expert move selected in given state. The network trained using 30 million positions from the KGS Go Server.
- The second stage is to improve policy network using reinforcement learning policy gradient (RL), by maximising the winning . The process start by initialising policy gradient weights using (SL) policy network, followed by games between the current policy network and and a randomly selected previous version of the policy network.
- The Final stage of the pipeline is to train value network which has a similar architecture to the policy network, but instead of predicting a probability distribution, it predict the winner of the games played by (RL) policy network against it self. The the network was trained by regression on state outcome pairs, using stochastic gradient descent to minimise the mean squared error between the predicted value, and the corresponding outcome.

The authors approach effectively combine reinforcement learning policy and value network with Monte Carlo tree search, which utilise random sampling of tree with evaluation of the game tree branch. The tree branches traversed by simulation, here each simulation traverses the tree by selecting the edge with maximum action value, and a bonus which depends on a stored prior probability for that edge.

The authors approach effectively combine reinforcement learning policy and value network with Monte Carlo tree search, which utilise random sampling of tree with evaluation of the game tree branch. The tree branches traversed by simulation, here each simulation traverses the tree by selecting the edge with maximum action value, and a bonus which depends on a stored prior probability for that edge.

To evaluate AlphaGo, the authors ran internal tournament among variants of AlphaGo , others open source Go programmer and state of the art commercial Go programs.

The result of the tournament was winning 494 games out of 495, around 99.8% of the games. Which show’s single Alpha Go is stronger then many existing Go program including state of the art.

The final challenge for AlphaGo is to play game against Human expert player. The game was played agains Fan Hui professional 2 dan and winner of the 2013, 2014 and 2015 European championships, where AlohaGo was the winner.