Review on AlphaGo: Deep Neural Networks with Tree Search

AlphaGo is the computer program developed by DeepMind of Google to play the game of Go, which is intractable by brute force computation. It has achieved and actually surpassed the level of best human players, a feat previously thought to be decade away. It uses 'value (neutral) network' to evaluate board positions and 'policy network' to select moves. A new search algorithm is also introduced which combines Monte-Carlo tree search with value and policy networks.

Methodology

a) Firstly, a supervised learning (SL) of policy network is trained based on deep convolution neural networks, using 30 million KGS positions from games of human experts. Their prediction of human expert action has much higher accuracy than other existing programs.

b) Secondly, a reinforcement learning (RL) of policy network is trained using games between current policy network and randomly selected previous iterations of the policy network. For each board state, it uses the reward function at terminal state (end of game) to update the weights of neutral networks. RL policy network typically has a much higher win rate than SL policy network.

c) Thirdly, RL of value network is obtained by predicting outcomes from positions of games, using same policy for both players. It has similar network structure as policy network, but outputs single prediction instead of a probability distribution. It uses regression on state-outcome (win/loss) pairs. Using KGS human expert positions alone may lead to overfit due to successive positions are strongly correlated. A new data set by playing RL networks and itself is used to greatly reduce the overfit.

d) Lastly, a Monte-Carlo tree search algorithm is developed in combination with policy and value networks. Each edge of search tree has action value and visit counts from all simulations. Each leaf node is expanded by SL policy network and evaluated by value network. At the end of simulation – completion of game, the action values and visit counts of all edges are updated. Once search is completed, the algorithm chooses the most visited move from the root position.

Conclusion

The novel design of policy network and value network in AlphaGo helps it develop superior heuristic on the board positions, therefore bypassing the difficulty of intractable computational complexity. The efficient Monte-Carlo tree search algorithm with the two networks in AlphaGo achieved dominant winning rate against other professional programs, defeated a human expert first and one of the best human player shortly afterwards. Although Go is in principle a complete information game, the way AlphaGo masters the game promises broader application of deep learning in non-complete information game.