Research Review

Mastering the game of Go with deep

neural networks and tree search
By: M.Shokry

SUMMARY

For a perfect search for a game agent to determine the best move to play, the agent will search across the all possible moves in the game with a factor of ~ p^d where in chess (b~35 and d~80), as deep blue used the computational power to search across the tree and defeated the world champion in the game, this tree is almost impossible in Go game as (b~250 and d~150),

So Deepmind combined the following strategies to search and evaluate acroos the game board:

1. Supervised Learning of policy network: A 13 layer policy network,The rule of this network is to return the probability of the maximum likelihood of a human expert move in the game of GO $p_\sigma(a|s)$ where s is a simple representation of the board state then the output feeded to a softmax layer,The results was 57% accurate .
2. RL of policy networks: Starting the training with the previous weights of the network and then stronger versions played with a randomly selected previous iteration to improve the accuracy and to overcome the overfitting, the RL won more than 80% of the games played with the SL policy network.
3. RL of value networks: It has the same structure of the policy network but the output of this network is binary is this play is good or bad to trim the bad moves from the search,using stochastic gradient descent to minimize the MSE. To overcome the overfitting a 30 million was generated by playing with the other iterations, the results was 0.226 and 0.34 MSE for the training and test sets respectively.
4. Searching with policy and value networks: AlphaGo combines the policy and value networks in an MCTS algorithm that selects actions by lookahead search. Evaluating policy and value networks requires several orders of magnitude more

computation than traditional search heuristics. To efficiently combine MCTS with deep neural networks, AlphaGo uses an asynchronous multi-threaded search that executes simulations on CPUs, and computes policy and value networks in parallel on GPUs.The final version of AlphaGo used 40 search threads, 48 CPUs, and 8 GPUs.

Results

1. *AlphaGo achieved a 99.8% winning rate against other Go programs, and defeated a human professional player in the champion by 5 games to 0.This is the first time that a computer program defeated a human professional player in the full-sized game og Go.*

Resources

[1] https://storage.googleapis.com/deepmind-media/alphago/AlphaGoNaturePaper.pdf

[2] https://www.youtube.com/watch?v=tXlM99xPQC8

[3] https://www.youtube.com/watch?v=vC66XFoN4DE