# Research Review

"Mastering the game of Go with deep neural networks and tree search" https://storage.googleapis.com/deepmind-media/alphago/AlphaGoNaturePaper.pdf

# Summary

## Techniques

They use *Value Networks* to evaluate board positions and use *Policy Networks* to select moves. And they skillfully combines:

1. **supervised learning** (SL) from human expert games
    1. train policy network fast and efficient updates
    2. also train a fast policy to be able to rapidly sample actions
2. **reinforcement learning** (RL) from games of self-play
    1. train policy network by optimizing the final outcome of games
    2. adjust the policy towards the correct goal of winning games
    3. train value network to predicts the winner of games played by the RL policy network against itself

with Monte Carlo Tree Search (MCTS).

### Supervised learning (SL) of policy networks

The SL policy network alternates between convolutional layers with weights. The input to the policy network is simple representation of the board state and a final outputs is a probability distribution over all legal moves. The policy network is trained on randomly sampled state-action pairs, using stochastic gradient ascent to maximize the likelihood of the human move selected in state.

### Reinforcement learning (RL) of policy networks

This training pipeline aims at improving the policy network by policy gradient reinforcement learning. They play games between the current policy network and a randomly selected previous iteration of the policy network. This randomize stabilizes training by preventing overfitting to the current policy. They use a reward function that is zero for all non-terminal time steps, and for terminal, reward is from the perspective of the current player at each time step +1 for winning and -1 for losing. Then weights are updated at each time step by stochastic gradient ascent in the direction that maximizes expected outcome.

### Reinforcement learning of valie networks

This stage focuses on position evaluation, estimating a value function that predicts the outcome from position of games by using same policy for both players. They approximate the value function with strongest policy (by RL policy network) using a value network with weights. The neural network has a similar architecture to the policy network, but outputs a single prediction instead of a probability distribution.

### Searching with policy and value networks

AlphaGo combines the policy and value networks in an MCTS algorithm, that selects actions by lookahead search. Each edge of the search tree stores an action value, visit count, and prior probability. The output probabilities are stored as prior probabilities for each legal action. The leaf node is evaluated in two very different ways: first, by the value network; and second, by the outcome of a random rollout played out until terminal step using the fast rollout policy; these evaluations are combined, using a mixing parameter lambda, into a leaf evaluation.

## Results

- AlphaGo archieved a 99.8% winning rate against other Go programs
- Defeated the human European Go champion by 5 games to 0