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

A brief summary of goals and techniques introduced

Game of Go due to its enormous search space is considered a challenge for AI, but Alpha Go is viewed as a breakthrough on this field, combining Value Networks to evaluate board positions and Policy Network to select moves. Both networks are deep neural networks which are trained a combination of supervised learning from plays of human experts, and reinforcement learning from self-played games.

In the paper a new search algorithm is introduced that combines Monte Carlo simulation with value and policy networks. Using this search algorithm, and was able to achieve winning rate of 99.8%, also defeated the human European Go champion by 5 games to 0.

It is not possible to use minimax to search the game space in such a game as Go, so instead as heuristic search method, Monte Carlo tree search has been utilized before, which uses Monte Carlo rollouts to estimate the value of each node in the tree, by having more games played the search tree becomes bigger, and meantime the node values become more accurate. previous works focused on policies to choose the nodes which try to narrow search to only high-probability nodes. Instead AlphaGo uses neural networks are uses, in similar fashion used in image classification to construct a localized representation of game.

Training of networks are conducted in three stages:

1. In supervised learning of policy networks for the first stage the networks are trained using randomly chosen state-action pairs from human experts games, also network has 13 layers. This helps the network to achieve better results of predicting the next moves (57% comparing the up to date of submission the best results of other approaches by accuracy of 44%
2. For Reinforcement learning of policy networks, same architecture is uses as the unsupervised version. In this case when this network is playing against previous network, it wins about 80% of time.
3. For Reinforcement learning of value networks focuses on position evaluation, estimating a value function by predicting the outcome from positions of games played by using policy p for both players.

AlphaGo uses policy and value networks in MCTS by selecting the action by lookahead search, anyway this requires way higher magnitude of computation than traditional heuristics, to overcome this challenge AlphaGo uses multi-threaded search with simulations are based on CPUs and for both policy and value network in parallel on GPUs, paper claims that final version used 40 search thread and 48 CPUS and 8 GPUs. Also, a distributed version of AlphaGo was introduced which uses higher CPUs and GPUs.

AlphaGo archives 99.8% of winning rates against other Go programs, including GnuGo, Fuego, Pachi,Zen, Crazy Stone and