Research Review

By **Manasi Kulkarni**

Mastering the game of Go with deep neural networks and tree search.
AlphaGo by the DeepMind Team

Introduction:

The game of Go has long been viewed as the most challenging of classic games for artificial intelligence due to its enormous search space and the difficulty of evaluating board positions and moves. After implementing our first game-playing agent it was very exciting to research and review the seminal paper by the DeepMInd Team "Mastering the game of Go with deep neural networks and tree search" that appeared in the journal Nature in early 2016. In the game of Go, where the branching factor is ~250 and a typical number of moves is ~150, the DeepMind team's goals was to find and apply novel techniques that are successful in games that have massive search spaces.

Besides the obvious main goal of reducing the search space of possible moves, it seems to us that the DeepMind Team was targeting for effective move selection and position evaluation functions for a Go program, based on deep neural networks that are trained by a novel combination of supervised and reinforcement learning which plays at the level of the strongest human players, there by achieving one of artificial intelligence's "grand challenges". "AlphaGo" was able to defeat the human European Go champion by 5 games to 0. [1]

Goals and System Design:

Three **policy neural networks** are used to decide which moves to investigate and which ones to play. They were trained to identify promising moves from a 19x19 image of the Go game board. Convolutional layers were used to construct a representation of the position. These neural networks are used to reduce the effective depth and breadth of the search tree: evaluating positions using a **value network**, and sampling actions using a **policy network**.

The **Supervised Learning (SL) policy network** is a 13-layer deep CNN trained on 30 million Go game positions. Given a game position, it predicts the next move, i.e. the 'most likely move'. It alternates convolutional layers followed by ReLU activations and is capped by a huge softmax that allocates a probability to each legal move.

The second stage of the training pipeline aims at improving the policy network by **policy gradient reinforcement learning (RL)**. It has the same structure as the SL network, but by making it play against itself 1.2 million times and beat earlier incarnations of itself, keeping the network weights of the winner, it became much stronger.

There is a third-policy network called the **Fast Rollout (FR) policy network**. It was trained to predict the next move like SL network, but it is a thousand times faster than the SL network. It is not as accurate, but because it is much faster, it is used to play out the rest of the game, predicting the most likely outcome following the predicted next move.

The **Value network** estimates the probability that the current position will lead to a win or a loss for the current player. When it was first trained, there was an issue of overfitting. To improve its ability to generalize, the AlphaGo Team trained it on the games collected during the reinforcement learning phase instead (~30M human games vs 1.5B self-play games).

The proposed AlphaGo program combines all these techniques with the MCTS (Monte-Carlo tree search) technique to achieve the record-breaking feat of beating a human Go champion. The use of deep neural networks for SL policy learning and value function evaluation contribute to the novelty of this work.

Results:

The AlphaGo program was evaluated against other Go playing programs that are based on high performance MCTS algorithms. Two versions of AlphaGo were considered, one that used a single machine (40 search threads, 48 CPUs and 8 GPUs) and another that was distributed across multiple machines (40search threads,1202 CPUs and 176 GPUs). Both versions of AlphaGo significantly outperforms previously existing Go-playing AIs. Single machine AlphaGo is many ranks stronger than any previous Go program, winning 494 out of 495 games (99.8%) against other Go programs. The distributed version of AlphaGo was significantly stronger, winning 77% of games against single machine AlphaGo and 100% of its games against other programs.

Apart from these, the AlphaGo program was also able to win a tournament, 5 games to 0, played across multiple days with Fan Hui, a professional go player and a 3-time European Go champion. [1] DeepMind's research has provided hope that, by similarly leveraging AlphaGo's novel techniques, human-level performance can be achieved in artificial intelligence domains that were also previously seen as unconquerable for a long time.

References:

[1] Mastering the game of Go with deep neural networks and tree search, by David Silver et al @ https://storage.googleapis.com/deepmind-media/alphago/AlphaGoNaturePaper.pdf
[2] AlphaGo Can Shape the Future of Healthcare, The Medical Futurist. http://medicalfuturist.com/alphago-artificial-intelligence-in-healthcare.