Research Review
Jared Bowden

Source literature
Silver, David et al. (2016). *Mastering the game of Go with deep neural networks and tree search*. Nature. 529 (7587): 484–489.

Go is an adversarial two-player strategy game wherein players adjust the position of pieces ("stones", 181 black, and 180 white) on a 19x19 grid board with the goal of encircling more board territory than their opponent. The game of Go is of particular interest to the field of artificial intelligence owing to the fact that:

1. Go is a game of perfect information, suggesting that perfect play is theoretically possible.
2. The number of moves and board positions possible in Go make brute force search paradigms computationally intractable.
3. Professional human Go players have historically outperformed their programmatic counterparts.

Collectively, these observations suggest that professional human performance is achieved through a nuanced series of operations native to computational medium (the brain), and a combination of as-yet-discovered search heuristics that may be gleaned through the course of experience. While the goal of this paper was not to reverse engineer human play, the parallels between the AlphaGo solution and human models of performance are undeniable. From a high-level, AlphaGo was able to achieve success by training a deep neural network on a series of professionally played and self-played games of Go. The end result of this training is not a set of explicit rules, rather, it amounts to a series of weight-based probability distributions that converge on the best possible action for any given board state. Much in the same way that humans use learned experience to reduce the total search space of Go to arrive at a high-performance solution, AlphaGo uses a series of experience-based "policy" and "value" networks to learn actions most conducive to success.

Previous attempts at a computational solution to Go have used repeated random sampling method like Monte Carlo tree search to evaluate the value state of a Go game tree, and have achieved a level of performance comparable to ameture play. As mentioned above, the AlphaGo approach differs from previous methods owing primarily to its implementation of deep neural networks. This implementation consists of 4 different phases. In phase one, supervised learning was used to train a 13-layer neural networks on a dataset consisting of human expert-play games (30 million positions, total). This "policy network" was used to compute the most likely expert-level move, given any particular game state. Using a holdout dataset, the performance of these predictions was calculated at 57% accuracy. In phase 2, the policy network developed in phase 1 was refined through the use of reinforcement learning and self-play. In this phase, the network was played against a random selection of previous versions of of the network The authors claim this random selection of "opponents" was sufficient to avoid overfitting, and improve the ability of the final network to generalize to novel scenarios. Through reinforcement training, values of +1 were assigned to game states conducive to winning, and -1 to games associated with a loss, with synaptic weights were adjusted towards successful states through gradient ascent. The network that resulted from this training was termed a "reinforcement policy network". In phase 3, the reinforcement policy network is trained to produce an estimated value function that may, in turn, be used to predict the next move. Here, the "value network" model is trained to minimize the error between the predicted outcome of a move, and the actual outcome of a move. The authors claim this implementation produced comparably accurate predictions to results based on Monte Carlo models, but with 15,000 times greater computational efficient. In the final phase (3) the policy and value networks described above were deployed to search within the Go game tree and return an optimal move. Eligible moves were used assessed by using policy and value networks to forecast the strength of downstream outcomes. The resulting scores could then be propagated back to current game state, and evaluated to select a move most conducive to success.

The performance of the AlphaGo approach was evaluated in two different ways. In comparison to other Go programs, AlphaGo appears to be unrivaled, achieving a 99.8% overall win rate. Similar success was realized through play against human professionals, where AlphaGo beat the current European Go Champion in 5 of 5 games.

In conclusions, the authors have demonstrated the the seeming insurmountable search space of Go can be tamed through the application of multiphase deep learning neural networks. By using these models to evaluate board position and select moves, superhuman programmatic performance is now possible in Go.