UDACITY ARTIFICIAL INTELLIGENCE NANODEGREE

Research review of AlphaGo

Daniel Vernon

April 15, 2018

1 INTRODUCTION

Google's seminal AlphaGo paper[Silver et al 2016] revolves around the highly complex Chinese game of Go. Even though Go is a game of perfect information, it has an enormous search space making exhaustive search infeasible. It also has fewer obviously logical moves than Chess, a game often compared to it. When interviewing professional Go players in preparation for this paper, Demis Hassabishe (DeepMind CEO) found that the players, when asked why they did a specific move, often answered, "It felt right" [Youtube Nature Interview]. With this considered, it is often referred to as an intuitive game; a concept that AI has found notoriously difficult. AlphaGo aims to be a generalised algorithm that can learn any game and this paper is a large step forward to achieving this vision.

2 GOALS AND TECHNIQUES

The goal for this paper is simple in comprehension; to develop a system that can consistently beat a human expert(professional) at the game of Go.

2.1 TECHNICAL OVERVIEW

AlphaGo passes a 19x19 image of the board's current state and then uses a deep neural network to construct a representation of the board that it can understand. The search space is then reduced using a value network along with sampling actions using a policy network. The network parameters still need to be calibrated to produce expert moves and this is where two different types of training simultaneously take place.

2.2 Policy networks

The goal of the fist stage of this policy network is to be able to predict what moves an expert would make using data from thousands of human expert games. The policy network alternates between weights and a non-linear rectifier which has been proven to be more effective than a linear rectifier [Mass et al(2013)]. Finally, a probability distribution of all legal moves is outputted. The policy network is trained to maximise the likelihood of the move selected being the human move, from the training data.

Once the supervised learning has taken place, AlphaGo moves onto reinforcement learning. Here the policy network takes the structure of the current iteration, including its weights, and then it plays games against randomly selected previous iterations of the policy network. They use a pool of iterations rather than just the most recent to avoid overfitting. A reward function is used to determine how good the moves are; in this case, a win or loss in the terminal node is given +1/-1 and then all weights are updated up the tree to maximise a positive output.

2.3 Value networks

The final approach AlphaGo uses in training is Reinforcement learning of value networks. This value function tries to predict the outcome of the board when both players are using the same policy network. For this network just one prediction is made and the weights are trained to minimise the mean squared error between the predicted value and the actual outcome of the game.

2.4 Searching networks

The policy and value networks are combined with a Monte-Carlo tree search (MCTS). An MCTS uses roll-outs to estimate the value of each in a search tree. Each time a simulation is ran, it becomes more accurate due to the tree expanding leading to children with higher values that can be selected.

3 Results

The results of AlphaGo versus every other high performance Go program were outstanding, achieving a win rate of 99.8% in 495 games. They also found when using a distributed computing set-up i.e using the processing power of multiple processors AlphaGo would beat itself 77% of the time. Finally, AlphaGo played the three-time European champion winner and won 5-0 in a clean sweep, which is the first time any program has beaten a professional Go player.

4 BIBLIOGRAPHY

Silver, D., Huang, A., Maddison, C.J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M. and Dieleman, S., 2016. Mastering the game of Go with deep neural networks and tree search. nature, 529(7587), pp.484-489.

Maas, A.L., Hannun, A.Y. and Ng, A.Y., 2013, June. Rectifier nonlinearities improve neural network acoustic models. In Proc. icml (Vol. 30, No. 1, p. 3).