Lucas Sabbatini de Barros Fonseca
Artificial Intelligence Nanodegree
November 23rd, 2017

Research on DeepMind's Mastering the game of Go with deep neural networks and tree search

The exponential nature of the search space for perfect information games makes exhaustive search infeasible for large games. For the game of Go, the time complexity, $O(b^d)$, reaches $\approx 250^{150}$. There are two general principles for reducing the effective search space: limiting the depth of the search by replacing the subtree below by an approximate value function and reducing the breadth of the search tree, by sampling actions from a policy $P(a|s)$ that is a probability distribution over possible moves a in position s. The DeepMind team presents AlphaGo, a new architecture for a search algorithm that successfully combines neural networks evaluations with Monte Carlo tree search (MCTS), introducing a novel combination of supervised learning (SL) and reinforcement learning (RL) in training that uses policy networks (for mapping states to probabilities for possible moves) and a value network (mapping states to evaluations).

The architecture does not consist on a single agent that both learns in training and sample actions in a game. Instead, in the learning procedure a pipeline is constructed so that agents learn from data and from each other. Some of these agents are then coupled together with a MCTS in a search algorithm for sampling actions in a game.

A supervised agent first learns from 30 million expert human moves from KGS Go Server, then its parameters are set as initial values for the parameters of another reinforcement learning agent, set to play against its previous versions so that it also learns long-term strategies. This trained agent, now with embedded expert moves and long-term policies, is set to play against itself 30 million times to generate data about games. (state, outcome) pairs are then used to train the value network, which will try to minimize the mean square error between the predicted value and the outcome reward of that game. A smaller network for rollout purpose was also trained with the expert human moves database.

For the search algorithm, the team coupled the value network and the two policy networks that were trained on KGS Go Server data, along with the value network, to a Monte Carlo tree search. When selecting a move, the tree is completely traversed n times without backup, simulating complete games. Each simulation runs as follows: at each time step t, the action a_t with the highest value for the sum of $Q(s_t, a)$ and $u(s_t, a)$, the former being average of evaluations of leaf nodes in simulations that s_t was part of and the latter the prior probability of action a decayed by repeated visits, to encourage exploration. The evaluation of the leaf node s_L will be a combination of the value network output for that state, and the output of a random rollout using the smaller policy network.

Due to computational requirements, two versions of AlphaGo were tested, a single-machine and a distributed one. In an internal tournament, the two programs played against variants of AlphaGo and several other programs, and showed superior strength against other Go programs, 99.8% wins for the single machine version, and 100% for the distributed version When played against Fan Hui, a professional 2 dan and winner of the 2013, 2014 and 2015 European Go Championship, in a five-game match, the distributed version of AlphaGo won the match and all games.

This novel approach of combining MCTS with policy and value networks for sampling and reinforcement and supervised learning for training created a system that can learn policies for actions and values for states, and use them to improve a search algorithm. Complications that arise solving the game of Go, such as exponential time complexity and heuristics too complex for direct approximation with neural network are commonplace in the artificial intelligence domain, hence the range of problems that may be solved, at least in a human level competence, using the presented strategy is vast.

Reference:

D Silver, A Huang, C J Maddison, A Guez, L Sifre, G van den Driessche, J Schrittwieser, I Antonoglou, V Panneershelvam, M Lanctot, S Dieleman, D Grewe, J Nham, N Kalchbrenner, I Sutskever, T Graepel, T Lillicrap, M Leach, K Kavukcuoglu, D Hassabis. "Mastering the game of Go with deep neural networks and tree search". *Nature.* 529. (2016): 484-503.