Michael (Jiashu) Zhou

Paul Eggert

CS35L Lab 5

TA: Ameya Kabre

Dec 8, 2017

AlphaGo: How Artificial Intelligence Mastered the game of Go

In the article "Mastering the game of Go with deep neural networks and tree search", author David Silver and Aja Huang analyzed the application of new technology in artificial intelligence for Go games. Go has long been viewed as the most challenging of classic games for artificial intelligence owing to its enormous search space and the difficulty of evaluating board positions and moves. However, the appearance of AlphaGo, a program that plays the board game Go, developed by Google DeepMind, has revolutionized people's perspective on artificial intelligence for Go games. With the recent development of the computer technology, a new approach to computer Go is introduced that uses 'value network' to evaluate board positions and 'policy network' to select moves. Also, a net search algorithm is introduced that combines Monte Carlo tree search with value and policy networks.

The deep neural networks used by AlphaGo are defined as convolutional networks. The convolutional network is a Sub-type of neural network that are especially well adapted for the processing of image data. It takes input as an image that goes through multiple layers of networks. At each layer in the network, a series of filters is applied on the image. It is highly computationally efficient on processing image data and it applies to all kinds of tasks that take image as input, such as digit, face, and license plate recognition. There are two different kinds of networks used in AlphaGo: policy network and value network. Both types of networks take input as the current game state, represented as an image. The value network provides an estimate of the value of the current state of the game. The input to the value network is the whole game board, and the output is a single number, representing the probability of a win. On the other hand, the policy network provides guidance regarding which action to choose, given the current state of the game. The output is a probability value for each possible legal move. Actions with higher probability values correspond to actions that have a higher chance of leading to a win.

Monte Carlo Tree Search is a method for making optimal decisions in artificial intelligence problems, typically move planning in combinatorial games. There are four processes involved in this algorithm, namely selection, expansion, simulation and backpropagation. Selection starts at root node, recursively select optimal child nodes and expand to different branches, representing different possibilities. If one root is a not a terminal node (I.e. it does not end the game) then create one or more child nodes. Then run a simulated playout until a result is achieved. Finally update the current move sequence with the simulation results. One important characteristics of Monte Carlo Tree Search is that it estimates the value of each state in a search tree. As more simulations are executed, the search tree branches out larger and the relevant values become more accurate. The policy used to select action is also improved over time, by selecting children with higher values.

Eventually, the policy converges to an optimal play, and the evaluations converge to optimal value function.

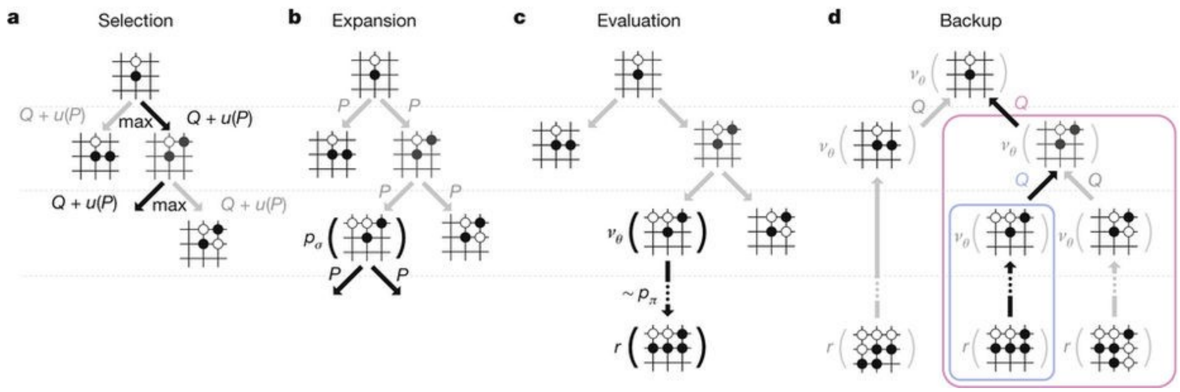

Figure 1: Monte Carlo tree search in AlphaGo

Knowing the neural networks and its underlying algorithm, the question is raised: How does AlphaGo make a move in a Go Game? It is accomplished by the invention called the novel form of reinforced learning. The net search algorithm is introduced that combines Monte Carlo tree search with value and policy networks. The deep neural networks are trained by a variety of new combination of supervised learning from human expert games, and reinforcement learning from games of self-play. Without look ahead search, the neural network play Go at the level of Monte Carlo tree search programs that simulate thousands of random self-played games. The system starts off with a neural network that knows nothing about the game of Go. It then plays games against itself, by combining this neural network with a powerful search algorithm. As it plays, the neural network is tuned and updated to predict moves, as well as the eventual winner of the games. This updated neural network is then recombined with the search algorithm to create a new, stronger version of AlphaGo, and the process begins again. In each iteration, the performance of the system improves by a small amount, and the quality of the self-play games increases, leading to more and more accurate neural networks and ever stronger versions of AlphaGo.

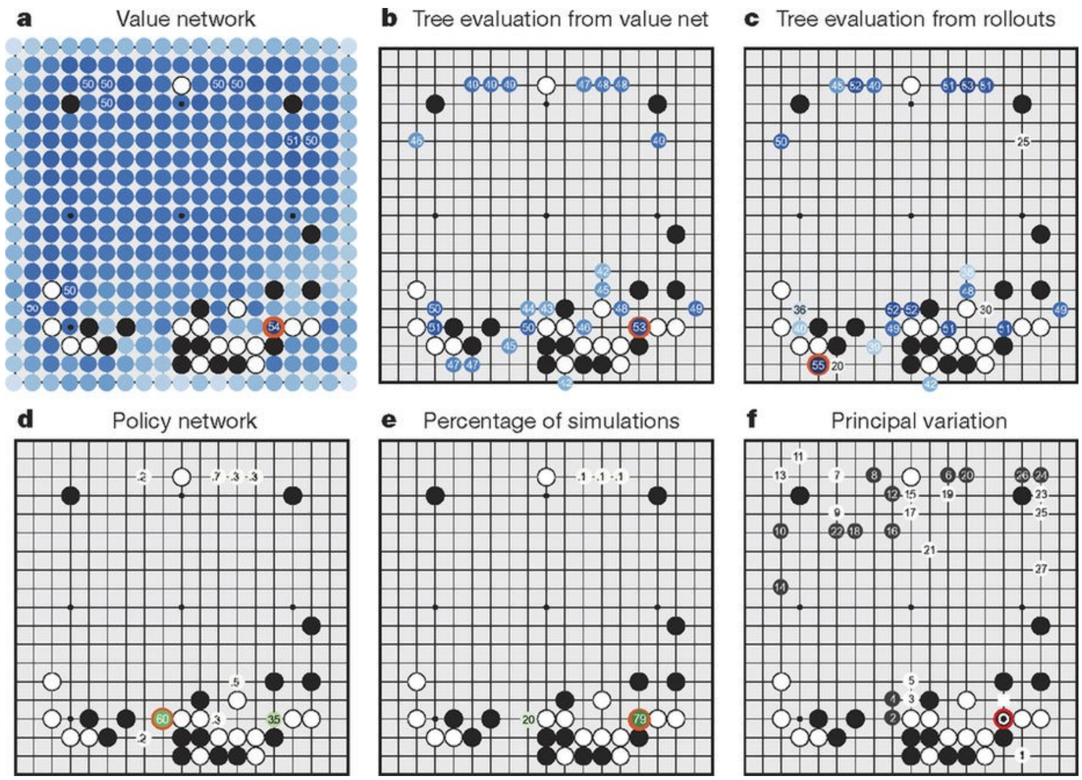

Figure 2: How AlphaGo (black) selected its move in a game

The success of AlphaGo has revolutionized artificial intelligence in the field of strategy board games. It is the first time ever that effective move selection and position evaluation functions for Go has been developed in artificial intelligence, based on deep neural networks that are trained by a novel combination of supervised and reinforcement learning. It is also the first time that a new search algorithm has been introduced that successfully combines neural network evaluations with Monte Carlo tree search. AlphaGo integrates these components together and becomes a powerful competitor against human players and dominated professional Go games recently. More importantly, the existence of

AlphaGo signifies the direction of future human artificial intelligence research and it also reveals the hidden power underlying AI across multiple fields that we have yet to discover.

Bibliography:

Silver, David, et al. “Mastering the Game of Go with Deep Neural Networks and Tree Search.”Nature News, Nature Publishing Group, 27 Jan. 2016, www.nature.com/articles/nature16961.

Burger, PhD Christopher. “Google DeepMind's AlphaGo: How It Works.” On Personalization and Data, On Personalization and Data, 6 Feb. 2017, www.tastehit.com/blog/google-deepmind-alphago-how-it-works/.

“AlphaGo Zero: Learning from Scratch.” DeepMind, deepmind.com/blog/alphago-zero-learning-scratch/.