

Mastering the game of GO with deep neural networks and tree search

Authors:David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel & Demis Hassabis.

Year: 2016

Journal: Nature

Report:

1. Abstract:

“The game of 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. Here we introduce a new approach to computer Go that uses ‘value networks’ to evaluate board positions and ‘policy networks’ to select moves. These deep neural networks are trained by a novel combination of supervised learning from human expert games, and reinforcement learning from games of self-play. Without any lookahead search, the neural networks play Go at the level of state-of-the-art Monte Carlo tree search programs that simulate thousands of random games of self-play. We also introduce a new search algorithm that combines Monte Carlo simulation with value and policy networks. Using this search algorithm, our program AlphaGo achieved a 99.8% winning rate against other Go programs, and defeated the human European Go champion by 5 games to 0. This is the first time that a computer program has defeated a human professional player in the full-sized game of Go, a feat previously thought to be at least a decade away.”

Introduction

- At the time that even chess takes an innumerable number of iterations to make the game win efficiently, there needs an either good hardware or algorithm to overcome the iterations of the game GO which takes an approximate of 250^{150} iterations which is thought to be impossible in the next ten years.
- But the team Alpha-Go came with an most efficient algorithm for the game to win even with the best players of the world. The process of forming and implementing the algorithm is the motivation of the paper.

Previous works

- There had been many works done on creating different games with artificial intelligence and gone through different stages in artificial game developments.
- Infact usage of deep convolutional neural networks, which the team had determined to use, had also been done on many of the applications from image classification to the atari game designs.
- To be precise very less work had been done to recreate the game go to function automatically because of the complexity it has. So, definitely it is a huge effort to accomplish it.

Content

- Every game algorithm have an optimal function which decides the optimal solution for every board game.
- Generally the board games are created virtually by “Monte Carlo Tree Search”(MCTS) which have a complexity of b^d .
- **Supervised Learning:**
 - a. This makes a way very complex procedure and needs a robust hardware to build the Go game virtually which would have an approximate breadth and width of 250 and 150 respectively.
 - b. There are some traditional ways of decreasing the breadth and width, but the team worked with combining the policies and the tree search to result in the optimum and fast solution.
 - c. The initial works in developing the game involved in combining the linear input features which do not result in optimal solutions often.
 - d. For this purpose they have created a fast policy which can sample actions during roll outs.
- **Reinforcement Learning:**
 - a. Then they have trained with the Reinforcement learning(RL) policy network to overcome the overfitting and maximizing the predictive accuracy.
 - b. Initially they have trained the program to predict the moves of the user with supervised learning.
 - c. The stochastic gradient ascent is used to expect the maximum probability of the move of the user which is followed by the training with Reinforcement learning.
 - d. They have created a 13 layer supervised learning policy network from KGS GO server, which predicted around 55%-55.7% moves.
 - e. The Reinforcement learning helped in overcoming the overfitting and optimizing the results. This network is done almost identical

to the supervised learning networks and weights are also almost same.

- f. Even in this the stochastic gradient ascent is used to maximize the probability and again increase the weightages of the neural networks by giving feedback.
- g. This procedure surprisingly resulted in 85% wins in self games compared to only 12% wins with supervised learning network.

- **Searching with the value and policy networks:**

- a. The policy networks are used to MCTS for the optimization of the solutions as it has much complexity.
- b. These networks in addition to the value networks helps in finding the efficient nodes, edges and their corresponding actions.
- c. A probability is taken for every expanding node and added as prior probability which is then added to the maximum action value.
- d. The overall maximum value results in the action of the program, which means that the coin is placed in the certain position of the board.

Results

- The game is then tested by playing with the most efficient online games like PACHI, FUEGO and GNUGO.
- It is also tested with the human champion of the Game go and the AlphaGO was said to win 5-0, which had been for the first time in the history to compete and with the human performance.
- They have used in total of 40 search thread, 48CPU's and 8 GPU's, according to the article, in which finally it took maximum of 5 seconds for each step.
- It wins in the most cases with other games with win percentage of 99.8 and also with 85% win when the handicapped coins were placed by the opponent.
- Even with the exception of rollouts, the AlphaGO had performed well with all other artificial Go games existed, with the assistance of the value networks, as an alternative approach to Montey Carlo search.
- It is really the best output of the algorithm with available resources and algorithms.

Conclusion

- In the many difficulties faced by the present day AI, Go is also one of them to be noted.
- It in fact had been achieved 10 years before than expected by many other researchers, which is a way satisfying and boosting the AI.
- It also increased the confidence of today's artificial intelligence to take the higher leaps with the available algorithms and also presented their efficiency when mixed with the new probabilities and possibilities.
- Finally, the paper greatly described their achievement with their process of implementing.