CPSC 540 Paper Review:
Modelling Human Behaviour in Board Games

Greg d'Eon, Shivam Thukral, Abner Turkieltaub

Mar 6, 2020

1 Relevant Literature

When searching for literature, we broadly looked for sources in 3 themes:

- **Existing AI for chess and Hanabi:** Understanding existing chess engines and Hanabi bots will be helpful, as we hope to use them to encode domain knowledge about these games. Any heuristics that they use might be useful for capturing important components of the game's state. There is little published work about the top chess engines, but information about them is available online:
  - Stockfish [1] is the strongest open-source chess engine that does not use any machine learning methods.
  - Leela Chess Zero [2] is also open-source, but it instead uses a deep neural network trained through millions of self-play games.
  - Other chess engines, such as Komodo [3], are also available. However, they are often closed-source, and most of them must be purchased.

  We are not aware of any publicly available Hanabi AI programs.
- **Machine learning for games:** We also looked for research-grade machine learning approaches to building AI for games in general, as we might hope to use or adapt many of their techniques.

  A variety of machine learning approaches have been used:
  - Furnkranz [4] reviewed many classic techniques for board game AI
  - AlphaGo [5] plays Go; it is bootstrapped from expert human play, then improved through self-play
  - AlphaGo Zero [6] also plays Go; it is trained purely from self-play
  - AlphaZero [7] plays Go, chess, and Shogi using a single, pure-self-play architecture
  - MuZero [8] generalizes these with one broader framework
- **Behavioural game theory:** On the more theoretical side, behavioural game theory research has studied how people make imperfect decisions. Qualitative findings from this area might be helpful to our project, helping us direct our modelling efforts:
  - Level-k models [9] describe the depth of each player's thought process: level-0 players pick actions randomly, and level-k players act as if their opponents are level-(k-1) players.
  - In quantal best response models [10], instead of picking the action that gives them the highest expected utility, people take a "softmax" over their actions.
  - Golman et al. [11] discuss how decision making is a process of accumulating evidence and how more "salient" actions might receive more attention during this process.

2 Paper Review

We chose to review Deepmind's original AlphaGo paper [5] for two reasons. First, it is a stand-out example of using modern deep learning techniques in board game environments. Second, it is most similar to our project, as the first policy network was trained on human play through supervised learning; later, they focused on learning purely from self play.

Summary of Contributions:

This paper describes AlphaGo, a state-of-the-art AI for playing the game of Go. AlphaGo consists of three parts. First, the *policy network* is a deep neural network that predicts which moves are the strongest in a given position. Second, the *value network* is a second neural network that takes a position and predicts the probability that each player will win. Third, AlphaGo selects moves using *Monte Carlo tree search* (MCTS), using the policy network to focus its effort on strong moves and the value network to predict the outcome of a game without needing to read many moves ahead. This combination of deep learning with MCTS algorithms extends on existing Go programs, which only used simple heuristic models for policies and values.

The policy and value networks were trained through a number of learning approaches. An initial supervised policy network was trained to predict expert players' moves, using a dataset of 30 million positions from online games. Then, the policy network was improved through reinforcement learning, playing self-play games against older policies and updating the weights based on the games' results. Finally, the value network was trained through supervised learning, predicting the winner on a set of 30 million new self-play games.

The results show staggering improvements over existing Go programs. First, the supervised policy network predicted expert human moves with an accuracy of 57%, significantly higher than the state-of-the-art's 44.4% accuracy. AlphaGo was also significantly stronger than all existing programs, winning 494/495 games against other programs and still winning a majority of games with a significant handicap. Finally, AlphaGo won 5 games in a row against the European champion, beating a professional Go player for the first time.

Strengths and Weaknesses:

The paper is extraordinarily clear. The problem is clearly motivated: at the time of writing, Go was an exemplary problem for AI, as playing Go well requires significantly more computation than other games such as checkers or chess. The technique is well-described and the methods used in the paper are clear. Finally, the evaluation is sensible, comparing AlphaGo against all of the best existing Go programs, and the results of this evaluation leave little doubt about AlphaGo's quality.

In our mind, the largest contribution is the combination of AlphaGo's deep learning components with existing MCTS algorithms. This paper is certainly not the first to attempt to use deep neural networks to predict moves in Go. However, what sets AlphaGo apart from existing methods is its ability to leverage these models in its tree search. In particular, the value network is less useful as a standalone model, but it is invaluable to the MCTS process: it allows the search to predict the outcome of a game without relying on high-variance rollouts, which can only play out the game according to a simple policy.

One key issue that is not mentioned in the paper is the deep networks' ability to generalize to new positions. The space of possible Go positions is enormous, and the training set used for the supervised policy network is biased towards the types of positions that appear in human games. It is unclear how drastically the policy network is changed through the self-play games. In any case, it appears quite possible that both the policy and value networks could overfit to their training data, making it possible to take AlphaGo "off-path" by playing unexpected, atypical moves. The authors do not appear to acknowledge or address this issue.

Lastly, the paper has some relevance to our project. Like AlphaGo's supervised policy network, we also hope to use machine learning to predict human moves in board games. However, there are some important differences. AlphaGo's only used the supervised learning process as a first step in building a stronger program, and so they focused entirely on games from strong players. In contrast, this supervised learning component is the core of our project, and we hope to use games from a variety of players, with a particular focus on identifying moves that weaker players make.

References

[1] Stockfish. `stockfishchess.org`. Accessed: 2020-03-06.

[2] Leela chess zero. `lczero.org`. Accessed: 2020-03-06.

[3] Komodo chess engine. `komodochess.com`. Accessed: 2020-03-06.

[4] Johannes Fürnkranz. Machine learning in games: A survey. *Machines that learn to play games*, pages 11–59, 2001.

[5] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484, 2016.

[6] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *Nature*, 550(7676):354–359, 2017.

[7] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419):1140–1144, 2018.

[8] Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, et al. Mastering atari, go, chess and shogi by planning with a learned model. *arXiv preprint arXiv:1911.08265*, 2019.

[9] Dale O. Stahl and Paul W. Wilson. Experimental evidence on players' models of other players. *Journal of Economic Behavior & Organization*, 25(3):309 – 327, 1994.

[10] Richard D McKelvey and Thomas R Palfrey. Quantal response equilibria for extensive form games. *Experimental economics*, 1(1):9–41, 1998.

[11] Russell Golman, Sudeep Bhatia, and Patrick Bodilly Kane. The dual accumulator model of strategic deliberation and decision making. *Psychological Review*, 2019.