The chess simulator based on the application of advanced artificial intelligence algorithms

Pavol Kacej, Juraj Gazda
Department of Computers and Informatics, Technical University of Košice, Slovakia

Faculty of Electrical Engineering and Informatics

MOTIVATION

As one of the first chess algorithms could be considered theoretical work of Alan Turing, started in 1947. However, the topic is still actual today. After Deep Blue's win over world champion Garry Kasparov in 1997, the chess engines are under constant improvement. In the recent years, it is possible to observe the rise of chess algorithms based of machine learning (ML), that are starting to overcome classical chess engines. The main principle of this thesis will be to explore existing chess ML algorithms, and create a solution combining detected advantages, all of than on standard hardware that is accessible to general public.

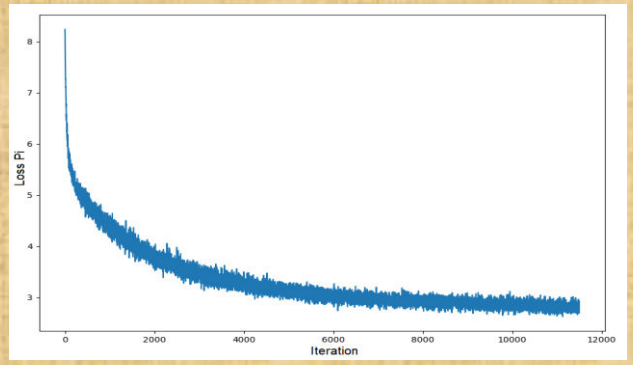
GOALS

- Analyze existing methods based ML for game of chess.
- Compare implemented ML algorithms via properly chosen game indicators.
- Suggest modification of the algorithms in order to improve chosen game indicators.
- Verify the solution by experiment.
- Implement user interface allowing user to compete against the implemented solution.

IMPLEMENTATION AND TRAINING

Multiple training approaches have been tested:

- The first approach was to use principles of AlphaZero, reinforcement learning technique used to learn the model via self-play. This learning process turned out to be considerably time consuming, and did not produce viable results on standard personal computer.



Learning curve over existing samples.

- In order to make the algorithm more time efficient, the second approach was introduced. It used existing training samples for supervised learning, rather than generating new samples via self-play. The model has been trained over 2 million pre-processed games. The games have consisted of real matches between professional chess players, and existing chess engines.

The model was able to reproduce variety of openings without any hard coding - Scandinavian Defense (left), English Defence (right).

- In order to adjust the move accuracy, and avoid mistakes in the middlegame and endgame, the model has been improved by adding search algorithm Minimax. With the help of the search, and heuristic system, the algorithm was better at finding moves that lead to winning material, or mating the opponent.

NEURAL NETWORK

The model is based on neural network with 4 convolutional layers, which makes it deep convolutional neural network (DCNN). The role of this DCNN is to recognize patterns over the 3D input (representing encoded chessboard), and produce the vector of probabilities for every possible move. The learning process consisted of 11500 iterations, each of them containing 2000 samples. The loss value has been continually decreasing from 8.24 to 2.71. After that, the training process has been terminated.

RESULTS AND CONCLUSION

The final solution has been showing excellent results in the sense of move accuracy. As it is visible in the pictures, the model was very accurate in the opening and middlegame, reproducing the standard openings and responding correctly to the openings. Due to help of Minimax algorithm and heuristics, the move precision increased especially in the endgame, as the algorithm was often able to keep the positive material balance, and search for winning moves. The solution was verified by experiments with human players and other chess engines, usually performing up to 90% of accuracy based on Stockfish analysis of depth 18, available on chess.com website.

The algorithm wins over human player with 89.6% accuracy.