Machine Intelligence

Project Report

ML model for Chess Game

Contribution From:-

1) **Vaibhav Vijay (PES1UG20CS479)**
2) **Prem Sagar J S (PES1UG20CS825)**
3) **Tushar J (PES1UG20CS472)**

Synopsis:-

Numerous published studies revealed that various researchers have attempted to build a program that learns to play cognitive games, given little or no earlier knowledge about the rule of game. Usual chess playing machine thoroughly explores the moving possibilities from a chessboard configuration to choose what the next best move to make.

The brute- force searching technique used by the Deep Blue chess engine has made vast impact in the ground of machine intelligence, but still found to be resource hungry.

Concept of Artificial Neural Networks presents a very simple and efficient approach to develop an intelligent chess engine which can assist and hint the possible move within the game using evolutionary and an adaptive computing technique on learning from human grandmasters.

Chess is known as the game of perfect information, since both players are conscious about entire situation of the game at all time and just by looking at the board it is considered as standard of testing intelligence.

The kind of victory of a machine over a human grand master has led to make chess one of the most used games to promote machine intelligence. The classical approaches to chess playing by machines have evolved over the years.

A two players game such as chess can be represented by a tree. Each node is a position on the board starting at root as beginning position of the game. The vertices are the possible moves which leads to next nodes. Minimax algorithm is the basic search tool used in most game programs. Minimax works by building a depth-first, left-to-right and search tree with alternating moves by Max and Min.

Chess and machine intelligence have shared a symbiotic relationship where one has enriched the other in multiple ways

Feasibility Study:-

Chess players exhibit superior cognitive ability and the skills needed to play chess have also been shown to correlate with several intelligence measures.

Chess methodology is all about a rich problem-solving domain and human's learning process was mimicked by studying an impressive number of chess games.

The disciplined and structured approach towards all documenting has also meant that Chess provides perfect opportunity to assess and understand the human mind's workings in countless situations created on a finite game like Chess on a chessboard. It is a finite game but with infinite moves.

By creating trained datasets, Machine Intelligence can help predict and analyse a player's growth based on the previous datasets of identically skilled players. Machine Intelligence provides a powerful tool to chess coaches, parents, educators and used to identify chess talent at a young age. If person's mistakes are predictable they get to know what to work on to hit the next level. Machine Intelligence systems have achieved superhuman performance and interact closely with human chess players both as opponents and preparation tools.

Description:-

we have used tensorflow linear classifier to train our model as it is well documented and very performant

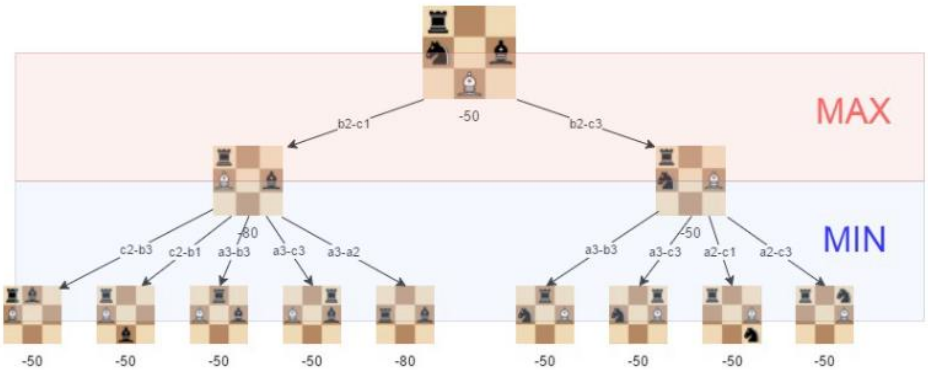
dataset-

1) The first 64 columns are a representation of the 64 squares of the current board.
2) Columns from 65 to 128 represent the starting position of a move: 1.0 if a piece was moved from that square, 0.0 otherwise.
3) Columns from 130 to 192 represent the ending position of a move: 1.0 if a piece was moved to that square, 0.0 otherwise.
4) The last column is the label: True if it was a good move, False otherwise

The model which we have trained will dismiss 50% of the possible moves given a board. It is done by finding the probability of a move being a 'good move'. All the moves are sorted according to this probability and only the 50th percentile remains.

We have used the minimax algorithm on the remaining moves. Depending on the complexity of the board, the search tree will go to a certain depth.

Minimax is a quite reliable algorithm and does good job at capturing enemy pieces. Applying Reinforcement learning instead of minimax algorithm is too complicated.

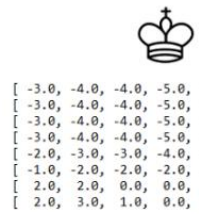


The minimax algorithm will explore the recursive tree of legal moves up to a certain depth and evaluate the leaves using an evaluation function. This algorithm returns the largest or smallest child's value to the parent node depending on the turn. This will allow to minimize or maximize the outcome's value at each level of the tree.

Evaluation function has two criteria:

1) The type of a piece. For instance, a knight is more valuable than a pawn.

2) The position of a piece. For instance, a knight is strong in the middle of the board, whereas a king would be vulnerable.



```
[ -3.0, -4.0, -4.0, -5.0, -5.0, -4.0, -4.0, -3.0],
[ -3.0, -4.0, -4.0, -5.0, -5.0, -4.0, -4.0, -3.0],
[ -3.0, -4.0, -4.0, -5.0, -5.0, -4.0, -4.0, -3.0],
[ -3.0, -4.0, -4.0, -5.0, -5.0, -4.0, -4.0, -3.0],
[ -2.0, -3.0, -3.0, -4.0, -4.0, -3.0, -3.0, -2.0],
[ -1.0, -2.0, -2.0, -2.0, -2.0, -2.0, -2.0, -1.0],
[  2.0,  2.0,  0.0,  0.0,  0.0,  0.0,  2.0,  2.0 ],
[  2.0,  3.0,  1.0,  0.0,  0.0,  1.0,  3.0,  2.0 ]
```

```
[ -2.0, -1.0, -1.0, -0.5, -0.5, -1.0, -1.0, -2.0],
[ -1.0,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0, -1.0],
[ -1.0,  0.0,  0.5,  0.5,  0.5,  0.5,  0.0, -1.0],
[ -0.5,  0.0,  0.5,  0.5,  0.5,  0.5,  0.0, -0.5],
[  0.0,  0.0,  0.5,  0.5,  0.5,  0.5,  0.0, -0.5],
[ -1.0,  0.5,  0.5,  0.5,  0.5,  0.5,  0.0, -1.0],
[ -1.0,  0.0,  0.5,  0.0,  0.0,  0.0,  0.0, -1.0],
[ -2.0, -1.0, -1.0, -0.5, -0.5, -1.0, -1.0, -2.0]
```

```
[  0.0,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0],
[  0.5,  1.0,  1.0,  1.0,  1.0,  1.0,  1.0,  0.5],
[ -0.5,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0, -0.5],
[ -0.5,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0, -0.5],
[ -0.5,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0, -0.5],
[ -0.5,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0, -0.5],
[ -0.5,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0, -0.5],
[  0.0,   0.0, 0.0,  0.5,  0.5,  0.0,  0.0,  0.0]
```

```
[ -2.0, -1.0, -1.0, -1.0, -1.0, -1.0, -1.0, -2.0],
[ -1.0,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0, -1.0],
[ -1.0,  0.0,  0.5,  1.0,  1.0,  0.5,  0.0, -1.0],
[ -1.0,  0.5,  0.5,  1.0,  1.0,  0.5,  0.5, -1.0],
[ -1.0,  0.0,  1.0,  1.0,  1.0,  1.0,  0.0, -1.0],
[ -1.0,  1.0,  1.0,  1.0,  1.0,  1.0,  1.0, -1.0],
[ -1.0,  0.5,  0.0,  0.0,  0.0,  0.0,  0.5, -1.0],
[ -2.0, -1.0, -1.0, -1.0, -1.0, -1.0, -1.0, -2.0]
```

```
[-5.0, -4.0, -3.0, -3.0, -3.0, -3.0, -4.0, -5.0],
[-4.0, -2.0,  0.0,  0.0,  0.0,  0.0, -2.0, -4.0],
[-3.0,  0.0,  1.0,  1.5,  1.5,  1.0,  0.0, -3.0],
[-3.0,  0.5,  1.5,  2.0,  2.0,  1.5,  0.5, -3.0],
[-3.0,  0.0,  1.5,  2.0,  2.0,  1.5,  0.0, -3.0],
[-3.0,  0.5,  1.0,  1.5,  1.5,  1.0,  0.5, -3.0],
[-4.0, -2.0,  0.0,  0.5,  0.5,  0.0, -2.0, -4.0],
[-5.0, -4.0, -3.0, -3.0, -3.0, -3.0, -4.0, -5.0]
```

```
[0.0,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0],
[5.0,  5.0,  5.0,  5.0,  5.0,  5.0,  5.0,  5.0],
[1.0,  1.0,  2.0,  3.0,  3.0,  2.0,  1.0,  1.0],
[0.5,  0.5,  1.0,  2.5,  2.5,  1.0,  0.5,  0.5],
[0.0,  0.0,  0.0,  2.0,  2.0,  0.0,  0.0,  0.0],
[0.5, -0.5, -1.0,  0.0,  0.0, -1.0, -0.5,  0.5],
[0.5,  1.0, 1.0,  -2.0, -2.0,  1.0,  1.0,  0.5],
[0.0,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0]
```

Literature Survey:-

Paper 1: The Impact of Artificial Intelligence on the Chess World

Abstract & Methodology:-

This paper focuses on key areas in which artificial intelligence has affected the chess world, including cheat detection methods, which are especially necessary recently, as there has been an unexpected rise in the popularity of online chess. Many major chess events that were to take place in 2020 have been canceled, but the global popularity of chess has in fact grown in recent months due to easier conversion of the game from offline to online formats compared with other games. Still, though a game of chess can be easily played online, there are some concerns about the increased chances of cheating. Artificial intelligence can address these concerns.

Conclusion & Motivation:-

It could be argued that chess is a battleground of artificial intelligence, as it is the perfect way to test the battle between human intuition and massive computing power. Due to the complex but well-defined nature of strategic games in general, they have proven to be a perfect environment for testing any progress in artificial intelligence.

Competing against other online players is a bigger part of chess competitions today than ever before, so is not surprising that online fair play is taken more seriously both by chess players who see cheating as a huge impediment to their entire game experience and by companies who create cheat detection software.

Cheat detection software has already been used for important games onboard, even before the COVID-19 pandemic, but this period of forced digitalization has meant that the implementation of cheat detection software has been done on a large scale, even for entertainment games.

Paper 2: Design and Development of Game Playing System in Chess using Machine Learning

Abstract & Methodology:-

Creating a powerful chess engine that can play at a superhuman level is not the hardest problem anymore with the advent of powerful chess engines like Stockfish, Alpha Zero, Leela chess zero etc. Most of these engines still depend upon powerful and highly optimized look-ahead algorithms. CNN (convolutional neural networks) which is used primarily for images and matrix-like data is been proved successful with games like chess. We treat here chess like a regression problem. Goal is to create a zero-knowledge chess engine. The trained model is then paired with minimax algorithm to create the AI. Our proposed supervised system can learn the chess rules by itself from the data. CNN can detect various tactical pattern to excel in games like chess even when using a limited lookahead search.

Conclusion & Motivation:-

Training the convolutional neural network to play chess using the publicly available data at FICS. The trained model that we got was able to detect patterns on the chessboard and was able to deliver various tactics to improve and win the game. We conclude that a better model is possible with more

refined and larger dataset having more high-level games. The model can be improved by giving more training time and can be benefited by using more better hardware. The network can be improved by using Residual Convolutional Neural Network in the neural net.

Paper 3: Artificial Neural Network Based Adaptive Chess Playing Machine

Abstract & Methodology:-

Chess is known as the game of “perfect information”, because both players are aware of the entire state of the game at all time, just by looking at the board, has since been

considered as standard of testing intelligence. Artificially intelligent programs have been developed by such computing majors as IBM, specifically for the purpose of playing chess, and after Deep Blue, defeated the world chess champion Garry Kasparov, machines have been consistently beating man in the rapid version of the game. This limitation in the power of the machine exists due to the fact that the decision tree based hardware and software takes too much time to search the exhaustive option set for the correct move.

Conclusion & Motivation:-

The ANN’s can be trained to learn moves as complex as those of chess games have been long debated upon. Their work brings out the fact that ANN’s can indeed be used for playing, teaching or helping human in playing chess. We believe that the same approach can be used to develop a full-fledged chess playing machine.

Paper 4: The Future of Chess-Playing Technologies and the Significance of Kasparov Versus Deep Blue

Abstract & Methodology:-

We argue that the recent Garry Kasparov v/s Deep Blue matches are significant for the field of artificial intelligence in several ways, including providing an example of valuable baseline benchmarks for more complex alternatives to contrast and justify themselves. We will also briefly summarize some of the latest developments on computer chess research and highlight how our own

work on a program called Chester tries to build on those developments to provide such justifications.

Conclusion & Motivation:-

Search engine uses a variant of B* to conduct high-level selective search, MTD(f)

to conduct the relatively shallow low-level probes, and extremely fast move generation and transposition table routines, so as to reach a competitive level required for meaningful evaluation. We call this program Chester (Chess Terminator). The main novelty of this approach is means of attempting to learn good interval-valued evaluation functions, suitable for B*, as opposed to relying on heuristics such as null-moves to generate the bounds. The emphasis is on using chess as a domain to help test the generality of our machine learning techniques being developed for real world tasks such as monitoring the NASA Space Shuttle.

Paper 5: End-to-End Deep Neural Network for Automatic Learning in Chess

Abstract & Methodology:-

Presenting an end-to-end learning method for chess, relying on

deep neural networks. Without any knowledge regarding the rules of chess, a deep neural network is trained using a combination of unsupervised pretraining and supervised training. The unsupervised training extracts high level features from a given position, and

the supervised training learns to compare two chess positions and select the more favorable one. The training relies entirely on datasets of several million chess games, and no further domain specific knowledge is incorporated. The experiments show that the resulting neural network is on a par with state-of-the-art chess playing programs, which

have been developed through many years of manual feature selection and

tuning. Deep Chess is the first end-to-end machine learning-based method

that results in a grandmaster-level chess playing performance.

Conclusion & Motivation:-

Presenting the first successful end-to-end application of machine learning in computer chess. Similarly to human chess masters, Deep Chess does not assign

numeral evaluation values to different positions, but rather, compares different positions that may arise, and opts for the most promising continuation. Having observed the playing style of Deep Chess, we note that it plays very aggressively, often sacrificing pieces for long term positional gains. This playing style resembles very much the playing style of human grandmasters. While computer chess programs have long been criticized for being materialistic, Deep Chess demonstrates the very opposite by exhibiting an adventurous playing style with frequent positional sacrifices.

Paper 6:- Design, Development and Evaluation of a chess game in a Ubiquitous Environment

Abstract :-

In the recent years we have seen an increased popularity in game development using Smartphones, which has provided an increasingly ubiquitous platform for designing games. In this paper we wish to investigate the use of a modern Smartphone's capabilities in game development by implementing and evaluating a classic game on the iPhone platform. We identify the limitations and possibilities that this field offers to the different aspect of game design.

Conclusions :-

In conclusion, we believe that the iPhone is a very promising platform, and mobile gaming in general will be a very important academic and commercial field. We would like to make use of more of the built-in technologies, such as the accelerometer to rotate the board, and we would like to try techniques to reduce the battery consumption of our program. For example, we would like to try setting the maximum FPS to 20, since it does not make a difference to the user, and see what difference it makes on battery consumption. But the most important networking aspect is the ability to connect to the Internet from the device. This means that we can create applications that can connect to a server that keeps statistics, or applications that let you play multiplayer games with hundreds of other players on the Internet, regardless of location and country. Our goal here was to follow the most commonly used chess protocol and connect with existing Internet Chess Servers. We can even possible for a game to dynamically download 3D models from the internet when loading, so we, as the programmer, could just change the file containing the model in the server,

and all players will therefore have the updated model, with no application updates required.

References:-

Research Paper1 -

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7759436/

Research Paper2 -

https://www.academia.edu/74047322/Design_and_Development_of_Game_PlayingSystem_in_Chess_using_Machine_Learning

Research Paper3 -

http://www.ijarcs.info/index.php/Ijarcs/article/viewFile/1641/1629

Research Paper4 -

https://www.aaai.org/Papers/Workshops/1997/WS-97-04/WS97-04-003.pdf

Research Paper5 -

https://www.cs.tau.ac.il/~wolf/papers/deepchess.pdf

Research Paper6 -

https://www.researchgate.net/publication/311456876_Design_Development_and_Evaluation_of_a_Chess_Game_in_a_Ubiquitous_Environment