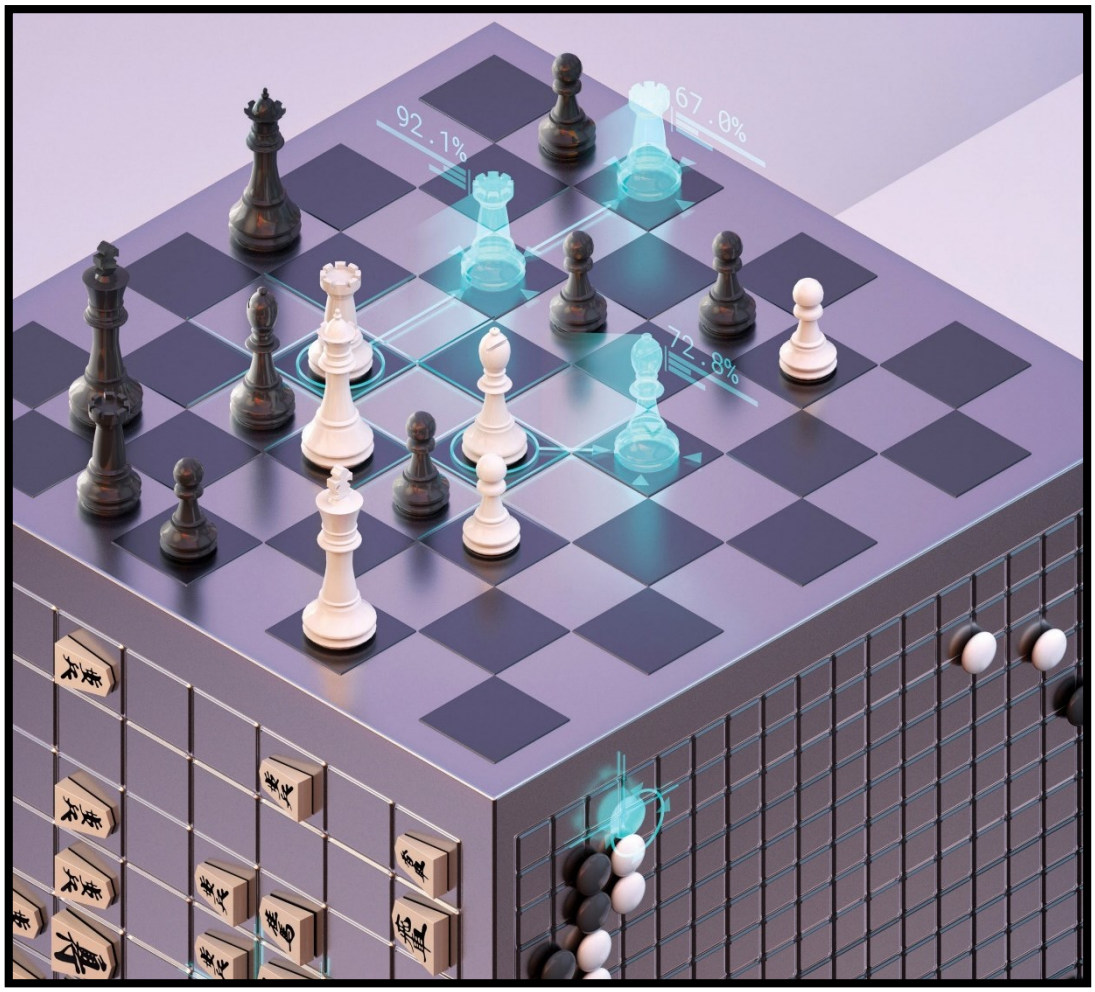
Open Programma Project Proposal

Chess AI



Speciation semester: ICT & Artificial Intelligence

Created by: Mickey Krekels

Class: AI45

Document Version: 0.4

Version management

Version	Date	Name	Description
0.1	14-06-2021	Mickey Krekels	Created the document structure and domain understanding
0.2	15-06-2021	Mickey Krekels	Created the Impact Assessment and Evaluation /deployment.
0.3	16-06-2021	Mickey Krekels	Finishing the proposal and added the conclusion
0.4	17-06-2021	Mickey Krekels	Fixed some of the grammar errors

Table of contents

Domain Understanding

Project goal

The goal of this project is to make an AI that plays the well-known game chess. That AI needs to be able to competitively play against a human player until the game is ended.

The possible drawback of choosing this kind of project is the long computing time while training. But this depends on the model techniques you use.

The Project Questions

Main questions:

Can an AI predict the next best possible move in a chess game?

Sub questions:

1. How to train the chess AI?
2. How do other chess engines do it?
3. What different model techniques are there?
4. What is the target group?

Context understanding

This part of the document will explain the idea and the possible problems I could face during this open programme challenge.

History of chess

The history of chess started in the country India, around 1500 years ago. It is the oldest known forerunner to the game and it was called **chaturanga** (see right image below), this was appropriated by the Persians, who edited the way it was played and called it **Shatranj** (see left image below). After the conquest of Persia in the 7th century by the Arabs, shatranj spread around the Caliphate (the rule or reign of a caliph or chief Muslim rule) before becoming known in Europe.

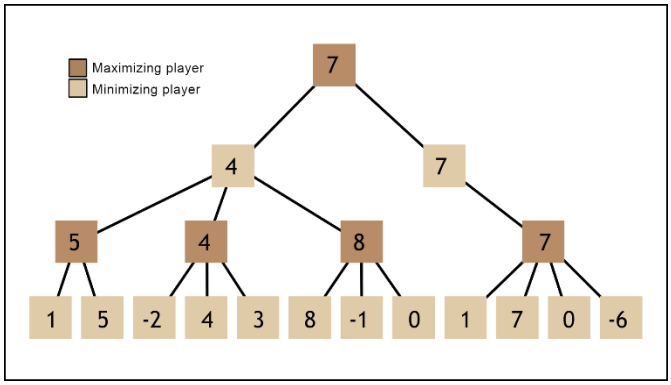
In the 15th century, the queen replaced the earlier vizier chess piece of the end in the 10th century and had become the most powerful piece, they think that this is the origin of most of the modern rules we play today.

On 10 February 1996 the first human against computer chess match began, in which Deep Blue (see image to the right) a chess computer became the first machine to win the game against a reigning world champion **Garry Kasparov**, some say that this marked the beginning of the AI era.

What are the different chess AI model techniques?

How do you get a computer to play chess? If you would want a simple program that plays against a human, you would make it that it makes random moves, but that in of its self is not very fun to play against. To make a good chess-playing computer algorithm, it requires finding out which moves are actually good and will lead to checkmate the king quickest. In theory, we could predict all the possible opponent moves and play the game based on that information, Unfortunately, this is not a realistic approach as the number of different piece location to keep track of grows exponentially, no computer has currently that amount of memory capacity to fulfil that task.

Instead, we define a certain depth for the AI possible current moves, for example, a depth of 5 will only look at 5 moves ahead of the opponent. But not every piece has an equal value on the board, sacrificing a pawn is better than losing a queen or a bishop piece. for that, we need an **evaluation function**. That when threatened it tries to save its most important pieces first, this is the same when attacking the competitor. To make that we give every own piece type a score (see left image below), the more important the piece the higher the score. We do the same for the opponent pieces but make it a negative number. Then to calculate the score we use a "**minimax search**", it chooses a node from a **branching tree** that produces the least amount of score loss, based on the lowest score loss value we make that move (see right image below).



This can be a big computing task for the pc but there are some ways to improve the calculation speed. The "**Alpha-Beta pruning**" technique is one of those ways, Alpha-beta pruning speeds up "**minimax search**" by skipping the unimportant nodes of a search tree. We don't need to search the branches of the node that had a lower score in the first tree layer, this massively improves the calculation speed.

Another technique is to learn a **neural network** to play chess, **deep learning** has a lot of big advantages, the algorithm is made based on how neurons work in your brain. If successful this might be the closest to a computer that behaves like a human playing a game of chess. But it is difficult to train and u need a lot of available data from previously played matches. There are still a lot of different ways to make a good model that learns how to play the game effectively, but these 2 ways are currently the two options I want to further explore in the machine learning phase.

How to train the model?

As I explained above, one of the most common ways is to use a point-based system and a search tree. We decide how deep the algorithm will look at possible future moves, the AI makes the next move based on that. By using this technique we could create a competitive AI for human players to play with. The other modelling strategy is creating a feed-forward neural network that makes decisions based on training with previously chess match data. Based on the best win score, I will choose that AI for the Deployment phase.

Are there examples of similar projects?

There are already perfect examples of chess models that play the game better than human players, for instance: Stockfish , Komodo or AlphaZero. There are even articles about neural network models such as Understanding AlphaZero: A Basic Chess Neural Network , DeepChess: End-to-End Deep Neural Network for Automatic Learning in Chess and Training a Convolutional Neural Network to Evaluate Chess Positions. So there is a lot of ways I can approach the ML phase during this project.

Impact assessment

Because of the 2 weeks timeframe, There is not a whole lot of time for documentation. For that reason, I decided to only do a quick scan for the Impact assessment of this project.

Quick Scan:

What is the challenge at hand? What problem (what 'pain') does this technology want to solve?

There is not necessarily a “pain” that this AI will solve, It is just made to play against humans. But if there is no other player available this could give the user a good and challenging competitor to play against.

In which way can this technology be used to break the law or avoid the consequences of breaking the law?

It could be used to cheat at chess games, with some events this could mee lots of reward money when winning the chess match. Other than that, no laws can be broken because of this technology.

Does this technology register personal data? If yes, what personal data?

It depends on what type of model I want to use for this project. The traditional algorithm trains based on playing against itself, so there is technically no human data needed. But if I want to use a neural network model approach then yes, saving the data to train the model is not necessary, but it is a good way of training the AI.

How does your technology affect the identity of users?

It does not affect the identity of the user unless they feel really bad after losing against a chess AI. In the case of **Garry Kasparov** the world chess champion of his time, He was very downhearted that he lost his title against a player that has circuits instead of a brain.

Who are the main users/targetgroups/stakeholders for this technology?

Stakeholder 1:

1.Name of the stakeholder

Professional chess players

2.How is this stakeholder affected?

They could use the technology to train with. This would be a great way for practising against a real player.

3.Did you consult the stakeholder?

No

4. Are you going to take this stakeholder into account?

Yes

Stakeholder 2:

1.Name of the stakeholder

Amateur chess players

2.How is this stakeholder affected?

They could use the technology to train with.

3.Did you consult the stakeholder?

No

4. Are you going to take this stakeholder into account?

Yes

Stakeholder 3:

1.Name of the stakeholder

Lonely senior

2.How is this stakeholder affected?

They could use this AI to play against, it can make for an interesting game of chess and they can play versus the model, whenever they want!

3.Did you consult the stakeholder?

No

4. Are you going to take this stakeholder into account?

Yes

Are you familiar with the fundamental shortcomings and pitfalls of data and do you take this sufficiently into account in your technology?

Yes, there are already lots of examples of AI that work better than mine and will probably ever be, but I am not aiming to reinvent the wheel in this project. There is a lot of reference material to be found online if I make a mistake, so I think I have most of the shortcomings and pitfalls covered for this project.

Does this technology have a built-in bias?

No, the AI can play against any player or other AI. There is no racial bias or other biases involved!

How is it explained to the users about how a technology works and how the business model works?

In the deployment phase of the project, here there will be extra context added where I explain how this AI was trained. So if the user desires it they can read this information.

What could possibly happen with this technology in the future?

The technology itself already was used to beat the world chess champion, Garry Kasparov, on February 10, 1996. So you could say we already passed that “future” point, but it did make a big impact on society. The loss of the Garry Kasparov title proved that the computer is capable of defeating humans, on human-like activities. Some say that this kicked off the AI area as we know it today!

Quick Scan Conclusion:

This project has a short time frame, therefore it is a shame that I could not include more research into the societal impact. But it has given me some ideas that I am going to use in the machine learning phase of this project. For example, I could save the data of the played games to improve my AI and I could also make an extra added page where I explain how the model works for the users that want to know. If I find the need to improve some of the Quick Scan answers (because for example I found new findings), I will certainly do that.

Evaluation and deployment

Is there a plan for domain knowledge verification?

The project “Chess AI” is not a new technology in the AI world, there are quite a few companies that have created a functional chess engine. So I know for a fact that the project is doable, also there are some tutorials about chess AI.

What are acceptable evaluation results?

If the AI can play against a human and make predictions that are human-like, than this would be an acceptable result.

Will the model be deployed? If so, how?

There are a couple of ways to deploy this technology, I could use the library Anvil. It creates a platform where you can test your model, by connecting your notebook with an API key the app can call functions from the Jupyter document. This would be a good deployment strategy.

I could also create my own API and host it for example on Asure. This would-be cool to make because I can create multiple chess front-ends in different frameworks and just call the API for the next move the AI should make.

Is there a feasibility deduction?

Yes, but it depends on the model type I want to use. Calculating the opponent next moves is difficult and can take a lot of computing time depending on how many steps it calculates. So a workaround for this problem is to drop the branches with the lowers score point in the decision tree.

Conclusion

In this proposal, I looked at the domain of chess, the future impact and the deployment. The goal of this document is to create a good starting point from where I will create this project. It gives me a lot of base knowledge that I need to make an effective model for predicting chess moves. For the next phase, I am going to use the models, search tree and neural networks. I already know how neural networks work, but the Domain Understanding in this document showed me how the train deep blue with search trees.

Therefore I think I am ready and prepared to start working on the next phase of machine learning!