# Algo de ajedrez (?)

April 26, 2024

Futuro Licenciado
futuro.licenciado@dc.uba.ar

Directores

Director1, Nombre          Director2, Nombre
director1@dc.uba.ar    director2@dc.uba.ar

# Contents

# 1 Feature sets (board encodings)

## 1.1 Half-Compact

$\langle piece\_rank, piece\_type, piece\_color \rangle \oplus \langle piece\_file, piece\_type, piece\_color \rangle$

$(8 * 6 * 2) * 2 = 192$ features

## 1.2 Half-Piece

$\langle piece\_square, piece\_type, piece\_color \rangle$

$64 * 6 * 2 = 768$ features

## 1.3 Half-Piece+Moves

HalfP $\oplus \langle move\_from, move\_to \rangle$

$768 + 64 * 64 = 4864$ features

Not friendly to efficiently update the network. It is almost always better to do a full refresh on eval.

## 1.4 Half-King-Piece

$\langle side\_king\_square, piece\_square, piece\_type, piece\_color \rangle$ excl. king

$64 * 64 * 5 * 2 = 40960$ features

There are variations to this feature set, such as HalfKAv2 or notably HalfKAv2_hm that is currently the latest feature set used by Stockfish 16.1. I will not consider them in this work.

## 1.5 Half-Relative(H|V|HV)King-Piece

$\langle side\_king\_file - piece\_file + 7, side\_king\_rank - piece\_rank + 7, piece\_type, piece\_color \rangle$ excl. king

$15 * 15 * 5 * 2 = 2250$ features (for HV)

only H or only V have $8 * 15 * 5 * 2 = 1200$ features

## 1.6 Half-Top(PP)

Statistical feature set, blabla, wasted features blabla

# 2 Efficiently updatable neural networks

NNUE (ƎUИИ Efficiently updatable neural network) is a neural network architecture that allows for very fast inferences. It was invented for Shogi by Yu Nasu in 2018.

In essence, NNUEs "Neural Network Update Efficient" are just regular neural networks that allow for really fast inferences.

... Most of the information described here can be found in Stockfish's documentation about NNUEs [1].

It is important to combine this with aggresive quantization techniques.

## 2.1 Architecture

arquitectura half, dos capas

## 2.2 Efficient updates

pesada al principio y liviana al final, acumular filas de la primera capa en domove, undomove

## 2.3 Stockfish quantization scheme

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. $\sin^2(\alpha) + \cos^2(\beta) = 1$. If you read this text, you will get no information $E = mc^2$. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. $\sqrt[n]{a} \cdot \sqrt[n]{b} = \sqrt[n]{ab}$. This text should contain all letters of the alphabet and it should be written in of the original language. $\frac{\sqrt[n]{a}}{\sqrt[n]{b}} = \sqrt[n]{\frac{a}{b}}$. There is no need for special content, but the length of words should match the language. $a\sqrt[n]{b} = \sqrt[n]{a^n b}$. Hello, here is some text without a meaning. $d\Omega = \sin\vartheta d\vartheta d\varphi$. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. $\sin^2(\alpha) + \cos^2(\beta) = 1$. This text should contain all letters of the alphabet and it should be written in of the original language $E = mc^2$. There is no need for special content, but the length of words should match the language. $\sqrt[n]{a} \cdot \sqrt[n]{b} = \sqrt[n]{ab}$. Hello, here is some text without a meaning. $\frac{\sqrt[n]{a}}{\sqrt[n]{b}} = \sqrt[n]{\frac{a}{b}}$. This text should show what a printed text will look like at this place. $a\sqrt[n]{b} = \sqrt[n]{a^n b}$. If you read this text, you will get no information. $d\Omega = \sin\vartheta d\vartheta d\varphi$. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression

of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language. $\sin^2(\alpha) + \cos^2(\beta) = 1$. Hello, here is some text without a meaning $E = mc^2$. This text should show what a printed text will look like at this place. $\sqrt[n]{a} \cdot \sqrt[n]{b} = \sqrt[n]{ab}$. If you read this text, you will get no information. $\frac{\sqrt[n]{a}}{\sqrt[n]{b}} = \sqrt[n]{\frac{a}{b}}$. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. $a\sqrt[n]{b} = \sqrt[n]{a^n b}$. This text should contain all letters of the alphabet and it should be written in of the original language. $d\Omega = \sin\vartheta d\vartheta d\varphi$. There is no need for special content, but the length of words should match the language. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. $\sin^2(\alpha) + \cos^2(\beta) = 1$. If you read this text, you will get no information $E = mc^2$. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. $\sqrt[n]{a} \cdot \sqrt[n]{b} = \sqrt[n]{ab}$. This text should contain all letters of the alphabet and it should be written in of the original language. $\frac{\sqrt[n]{a}}{\sqrt[n]{b}} = \sqrt[n]{\frac{a}{b}}$. There is no need for special content, but the length of words should match the language. $a\sqrt[n]{b} = \sqrt[n]{a^n b}$.

This is text bla bla. [2]

More testing on section 2.3.

**Rango de activación**: en el modelo original usamos ClippedReLU, asi que queremos que el rango vaya de 0..1 a 0..127.

Siendo $\boldsymbol{x}$, $\boldsymbol{w}$ y $\boldsymbol{b}$ los parámetros de una capa lineal sin cuantizar e $\boldsymbol{y}$ la salida de la misma, se tiene que:

$$
\begin{aligned}
\boldsymbol{y} &= \boldsymbol{x}\boldsymbol{w} + \boldsymbol{b} \\
s_a s_w \boldsymbol{y} &= (s_a \boldsymbol{x})(s_w \boldsymbol{w}) + s_a s_w \boldsymbol{b}
\end{aligned}
\tag{1}
$$

$$
s_o((s_a \boldsymbol{x})(s_w \boldsymbol{w}) + s_a s_w \boldsymbol{b}) = s_a s_w s_o \boldsymbol{y}
$$

## 2.4 Network sparsity

o combinar con 3.2? poner graficos con la sparsity de cada feature set, decir que es muy esparso todo y que se podría mejorar aún más

---

# 3 Training

## 3.1 Dataset

Lichess is a free online site to play chess, and it provides a database[1] with all the games played on the site. It consists of serveral compressed PGN files[2] splitted by month since 2013, that add up to 1.7 terabytes compressed. It contains over 5 billion games, that equates to around 200 billion positions. In reality, that many positions are too much to handle so I'll use only a fraction of them, but I restrict derived datasets to only take one sample per game, to maximize the diversity of positions.

## 3.2 Methods

### 3.2.1 Stockfish evaluations

### 3.2.2 PQR triplets

This is an additional technique I wanted to try, described in [METER REF BLOG]. Remember that we are trying to obtain a function $f$ (the model) to give an evaluation of a position. The method is based in the assumption that players make optimal or near-optimal moves most of the time, even if they are amateurs.

1. For two position in succession $p \rightarrow q$ observed in the game, we will have $f(p) \neq f(q)$.

2. Going from $p$, not to $q$, but to a *random* position $p \rightarrow r$, we must have $f(r) > f(q)$ because the random move is better for the next player and worse for the player that made the move.

   ... With infinite compute, $f$ would be the result of running minimax to the end of the game, since minimax always finds optimal moves.

## 3.3 Metrics

asd [1]

---

# 4 Engine implementation

## 4.1 Alpha-Beta

## 4.2 Prunes

# 5 Results

## 5.1 Active neurons

medir si hay feature sets que no usen neuronas, que esto disparo el uso de HalfTopK

# 6 Final words

## 6.1 Conclusions

## 6.2 Future work

Prueba de cita [1].

# References

[1]  Yu Nasu. "NNUE: Efficiently Updatable Neural-Network-based Evaluation Functions for Computer Shogi". In: *Ziosoft Computer Shogi Club* (2018). URL: `https://www.apply.computer-shogi.org/wcsc28/appeal/the_end_of_genesis_T.N.K.evolution_turbo_type_D/nnue.pdf`.