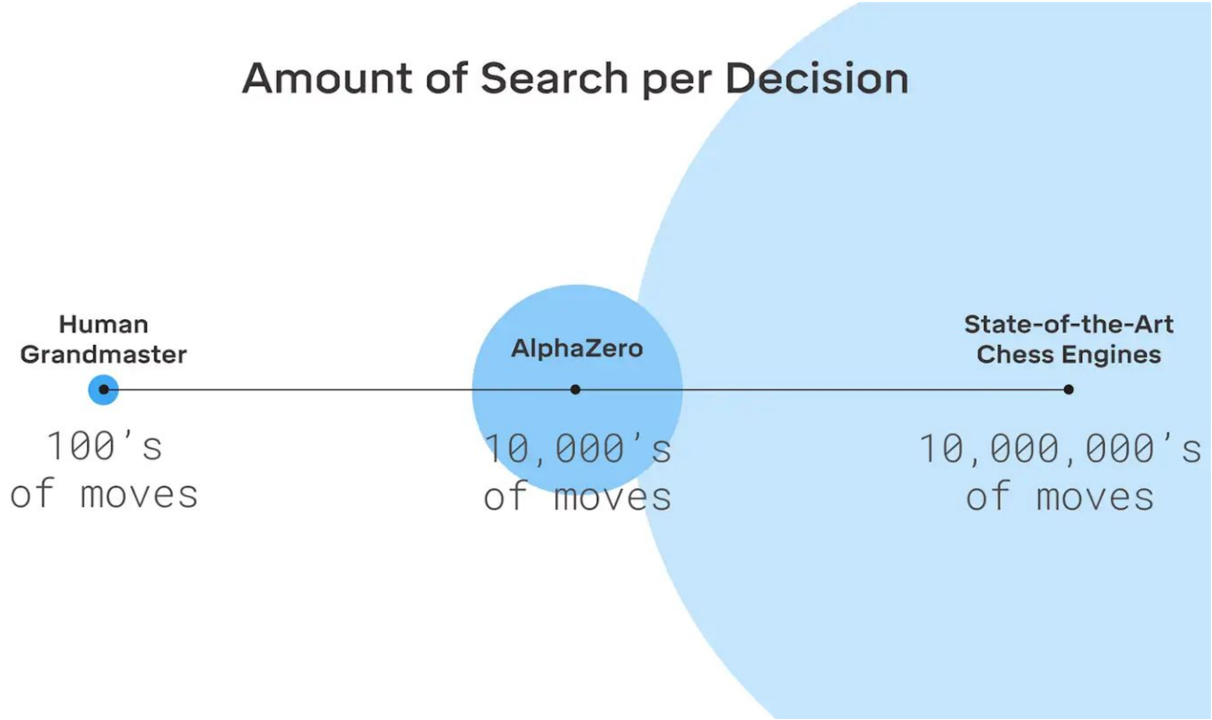
Chess position representations in AI and humans

Superhuman AI and humans search very differently

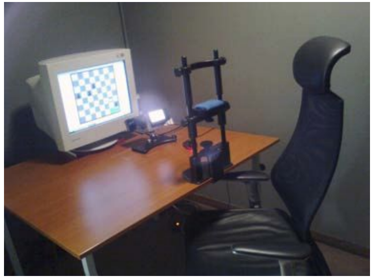


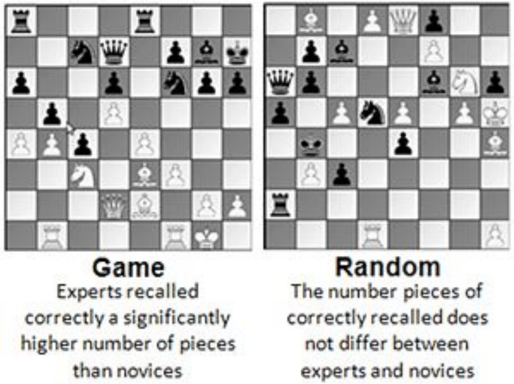
- Unlike human, computer computation is very cheap
- The latest neural nets can perform on a Grandmaster level without so much search, although one could argue that some of it is done implicitly

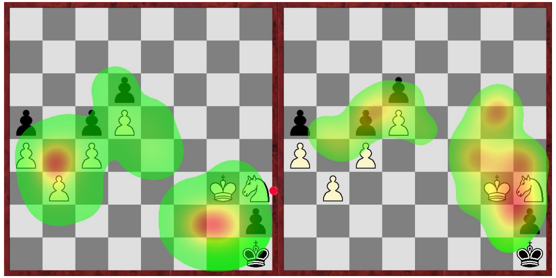
source: https://deepmind.com/blog/article/alphazero-shedding-new-light-grand-games-chess-shogi-and-go

We “know” how engines play chess, what about humans?

- Eye movement studies
- Recall of random and distorted positions
- Identification of chunks
- Research on chess problem solving
- ...







source: Hristova et al 2014, de Groot 1978, Gobet and Simon, 1996

Chunking

Psychologists seem to know a chunk when they see one. A definition, however, is hard to come by. (Terrace 2001).

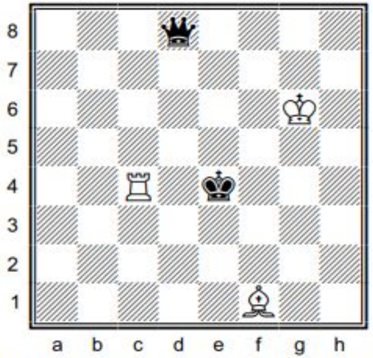


Figure 5.2: A chessboard with five pieces

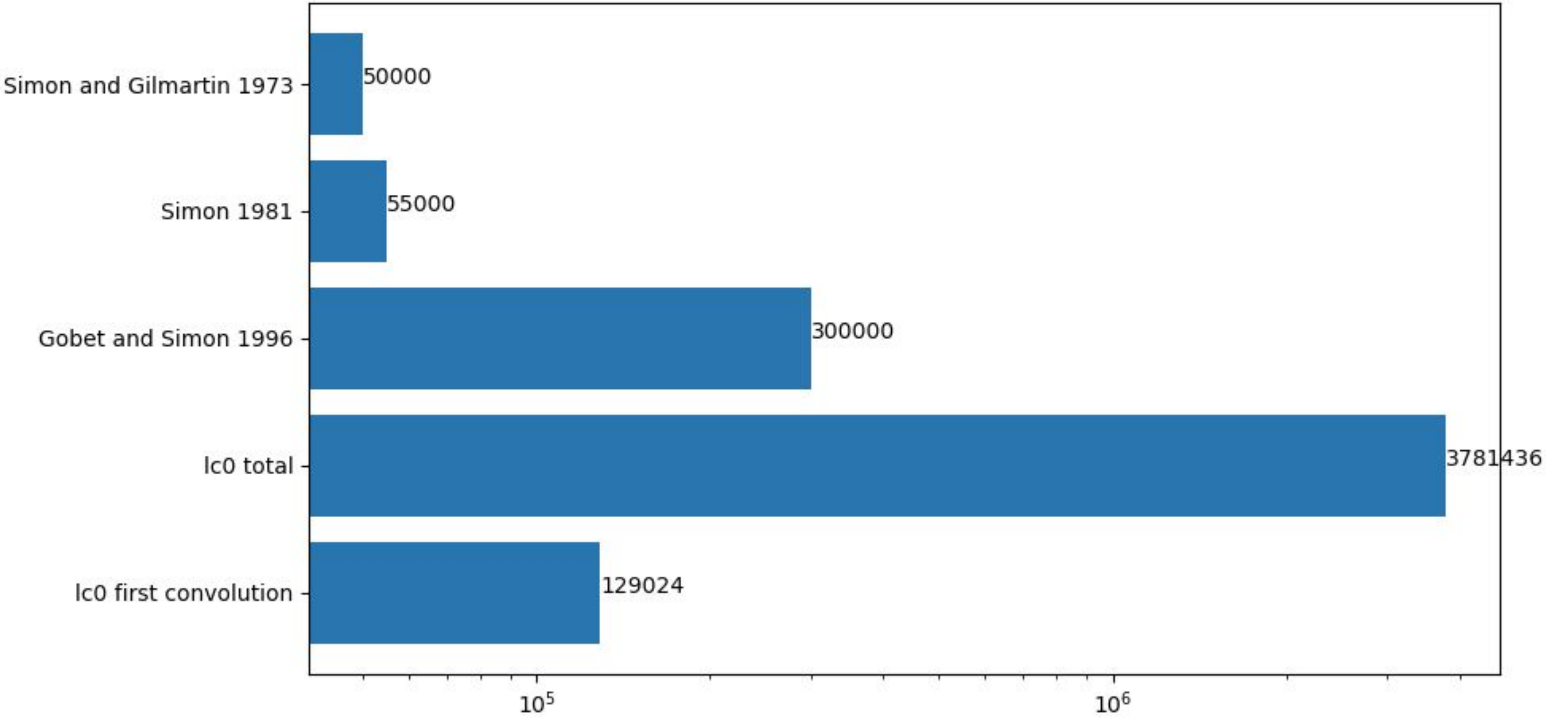
<qd8>
<ke4>
<ke4, qd8>
<Kg6>
<Kg6 ,qd8>
<Kg6, ke4>
<Kg6, ke4, qd8>
<Rc4>
<Rc4, qd8>
<Rc4, ke4>
<Rc4, ke4, qd8>
<Rc4, Kg6>
<Rc4, Kg6 ,qd8>
<Rc4, Kg6, ke4>
<Rc4, Kg6, ke4, qd8>
<Bf1>
<Bf1, qd8>
<Bf1, ke4>
<Bf1, ke4, qd8>
<Bf1, Kg6>
<Bf1, Kg6 ,qd8>
<Bf1, Kg6, ke4>
<Bf1, Kg6, ke4, qd8>
<Bf1, Rc4>
<Bf1, Rc4, qd8>
<Bf1, Rc4, ke4>
<Bf1, Rc4, <ke4, qd8>
<Bf1, Rc4, Kg6>
<Bf1, Rc4, Kg6 ,qd8>
<Bf1, Rc4, Kg6, ke4>
<Bf1, Rc4, Kg6, ke4, qd8>

Figure 5.3 Combining five chess pieces

“Practical” definition for chess - a chunk is a frequently occurring pattern

source: Cook 2011

How many chunks do chess experts use? And AI?



source: myself

How similar are human and chess representations?

- Representations are used to solve problems. Better representations are such that help to solve problems better.
- Both humans and computers make mistakes, so
- Can chess engine representation be used to estimate how difficult a chess task is?

First experiment

Data:

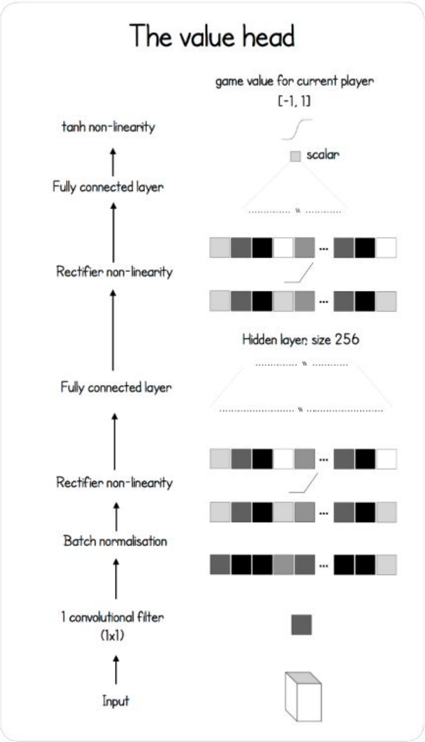
~50000 chess exercises (positions where 1 move is best and winning) with rating (the higher the rating, the more difficult the exercise)

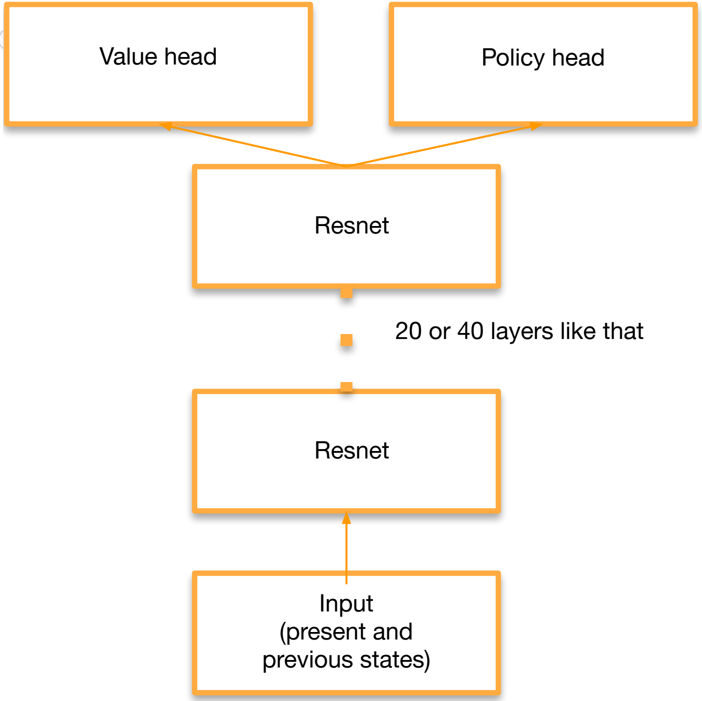
Experiment setup:

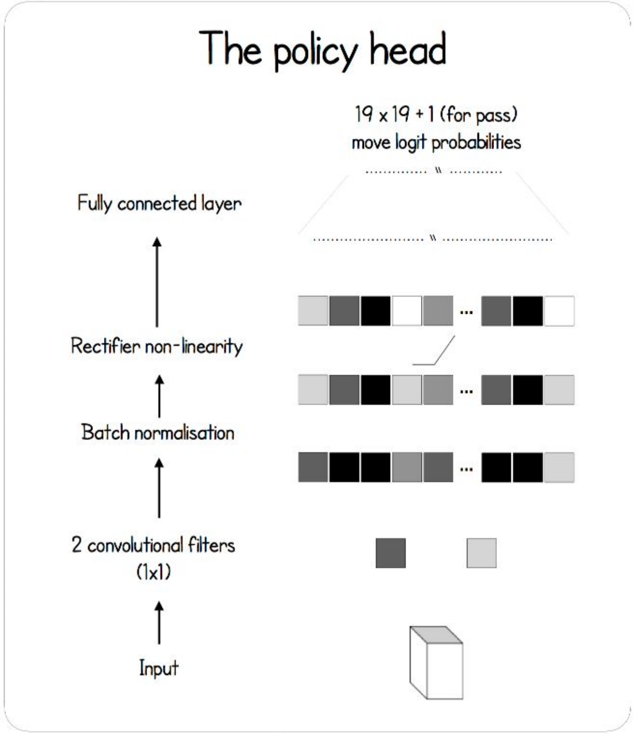
Use a SOTA neural chess engine to calculate activations at each layer, see if those activations make for good feature for a classification model that predicts whether the exercise is harder or easier than median

Expected result: after each layer, the results should be better and better until some point, when the features become specific for value/policy heads and no longer as useful

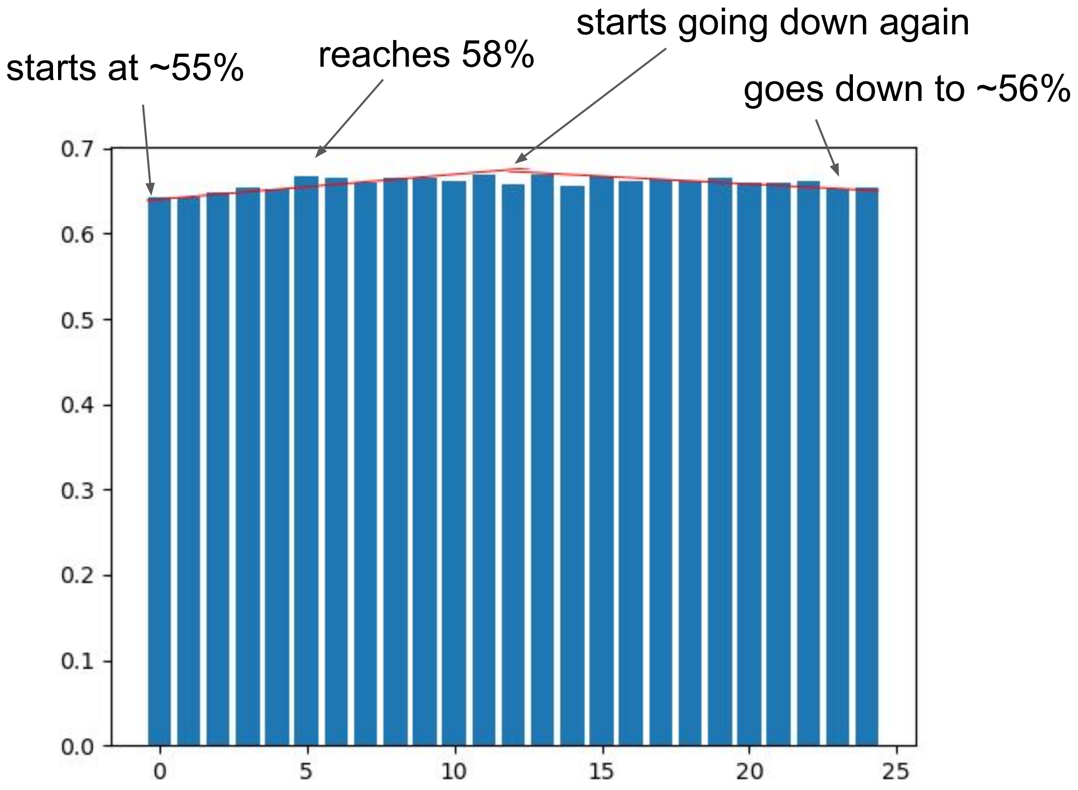
MCTS - RL engines







Result: the effect is there, but it’s not very strong



source: myself

Maybe “task difficulty” of a chess task is less related to evaluation and more to search?

Second experiment

Data:

~400 chess exercises (positions where 1 move is best and winning) with rating (the higher the rating, the more difficult the exercise)

Experiment setup:

Use a small neural chess engine to repeatedly try to solve the exercise with different number of nodes. Calculate success rates as well as average search tree statistics. Use those features to predict exercise difficulty as previously

source: myself

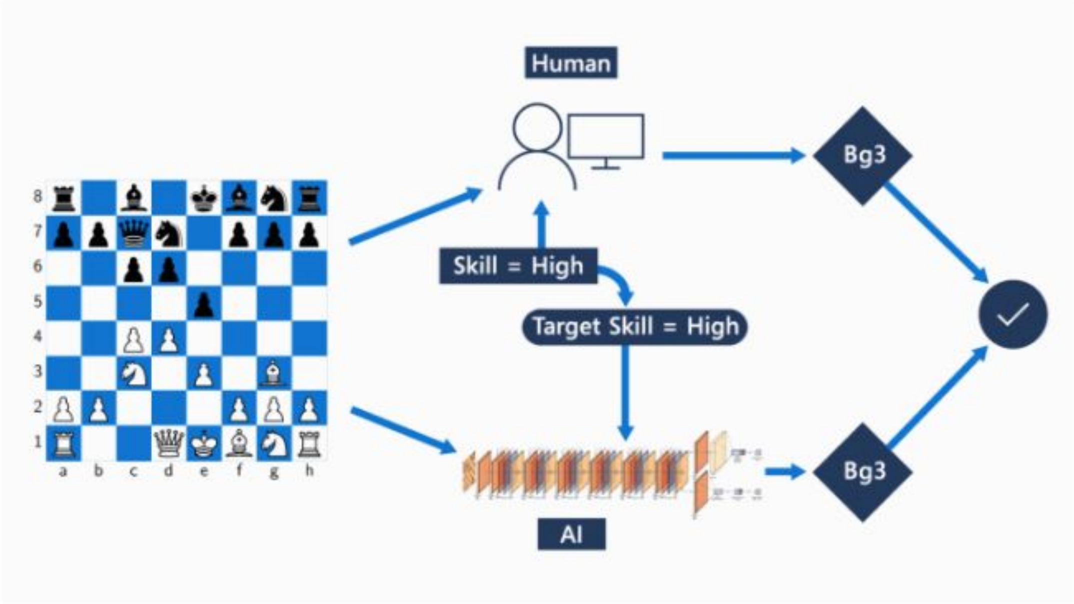
Results

10-fold cross validation gave **85% accuracy** on the prediction task with 5% standard deviation

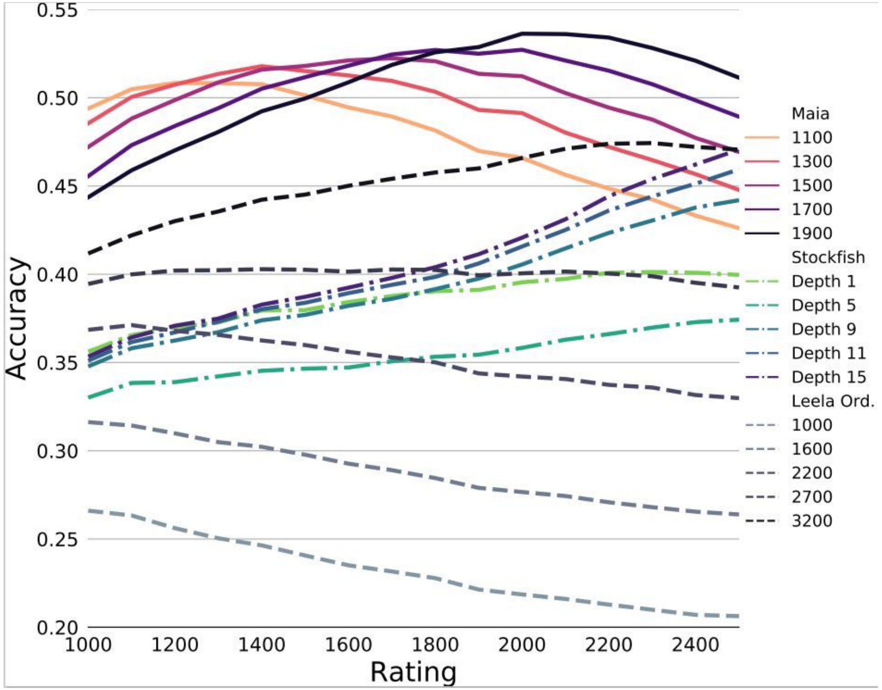
source: myself

What are others working on?

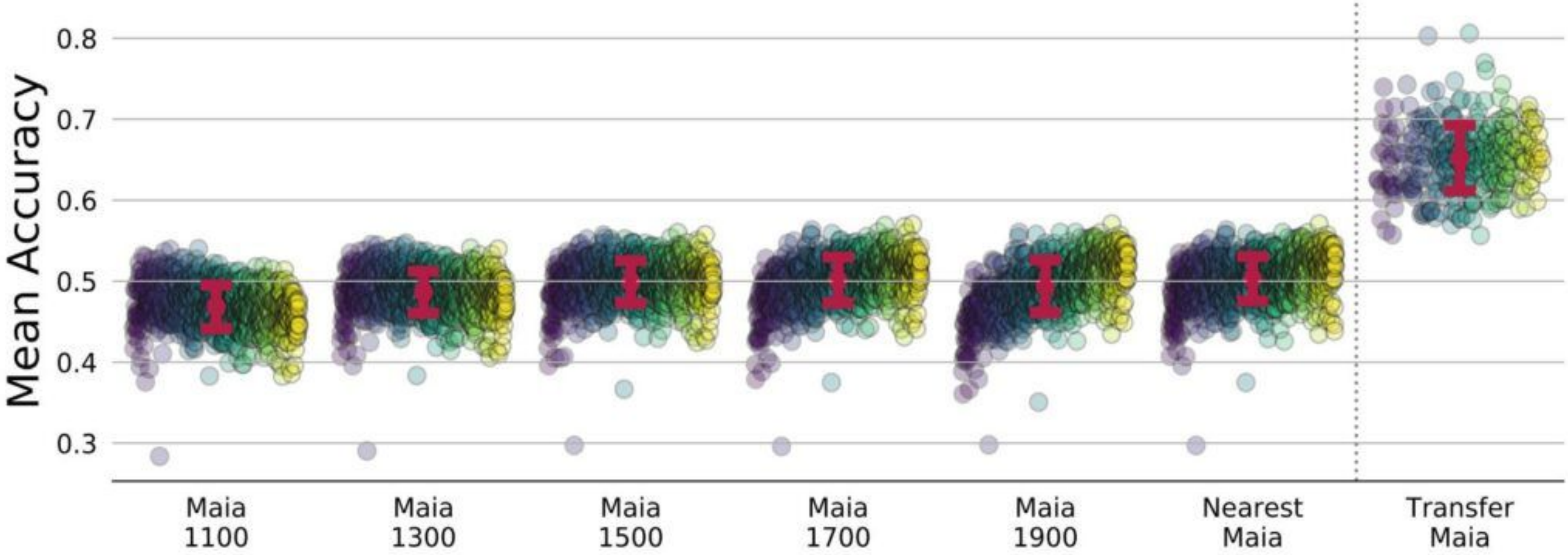
“Aligning Superhuman AI with Human Behavior: Chess as a Model System”



“Aligning Superhuman AI with Human Behavior: Chess as a Model System”



“Learning Personalized Behaviors of Human Behavior in Chess”



Thank you for your attention!
Questions?