Identifying Concepts Used by Human-Like Neural Network Chess Engines

Issac Li (issacli@princeton.edu)[1], Evan M. Russek (evan.russek@hunter.cuny.edu)[2], Ionatan Kuperwajs (ikuperwajs@princeton.edu)[1] & Thomas L. Griffiths (tomg@princeton.edu)[1,3]

[1]Department of Computer Science, Princeton University Princeton, NJ, United States
[2]Department of Psychology, Hunter College, CUNY, New York, NY, United States
[3]Department of Psychology, Princeton University Princeton, NJ, United States

Abstract

As neural networks reach expert-level performance in many domains, there is growing interest in understanding the internal representations that guide their decisions. Models trained to mimic human decisions, rather than strive for optimal performance, offer a particularly interesting testbed for interpretability. Identifying what information these networks encode can generate hypotheses about what information humans rely on when performing similar tasks. Here, we use concept-based interpretability methods from prior work on superhuman-level chess neural networks to study neural networks trained to emulate human play across differing skill levels. We find that human-interpretable chess concepts are decodable from their latent representations. For many concepts, decodability increases with human-emulated skill level, and also varies with network depth: simpler concepts peak early, whereas complex concepts emerge later. Together, these results show that interpretability tools can be applied to AI systems that mimic human behavior and suggest how task representations may shift with expertise.

Keywords: chess; artificial intelligence; neural networks; interpretability

Introduction

Neural networks are used as the backbone of many modern artificial intelligence (AI) systems, including large language models, translation, and object detection (LeCun et al., 2015; Sutskever et al., 2014; Vaswani et al., 2017). Given their increasing prevalence, recent work has attempted to understand how these networks function. This question of neural network interpretability has been investigated in domains such as computer vision, language, and chess (McGrath et al., 2022; Olah et al., 2020; Senel et al., 2018).

Beyond their applications in AI, another area of research has trained neural networks to mimic human behavior rather than optimizing for absolute performance. Such models are useful because they can help generate hypotheses about human cognition (Horton, 2023; Ji-An et al., 2025; Wang et al., 2023). Since directly measuring a person's internal processes as they make decisions is challenging, these models and their underlying computations can be used to characterize what those internal processes might be. This is particularly interesting for models that can approximate human behavior across a range of skill levels, offering ideas for how the information that people represent might change as they acquire expertise in some domain.

In this paper, we apply interpretability methods to understand the concepts used by neural networks that mimic human behavior in the domain of chess. Chess has been used as a foundational setting for studying complex planning and decision-making in both cognitive science and AI (Chase & Simon, 1973; Gobet, 2018; Newell, 1955). Recent advances have resulted in neural network systems that far outperform the best human players (Silver et al., 2018). Building on work analyzing human-interpretable concepts used by these superhuman chess systems (McGrath et al., 2022), we decode the concepts used by a system trained to mimic human behavior (McIlroy-Young et al., 2020).

We first show that human-interpretable chess concepts are present in models that emulate human play, and that the presence of these concepts is influenced by the strength of the players whose moves the model is predicting. We also highlight how the strength of each concept is related to model layer, with more complex concepts increasing in saliency in later layers and simpler concepts decreasing in saliency in later layers. Using the models as a proxy for human players, our results suggest the play of more skilled players seems to be better informed by various chess concepts compared to their less-skilled counterparts.

Background

Superhuman chess engines

Chess engines have long exceeded the performance of even the top chess players (Campbell et al., 2002). These engines are able to achieve superhuman performance due to their ability to evaluate an enormous number of potential chess positions effectively. Traditionally, such systems evaluated positions using handcrafted functions that are based on various heuristics derived from human play as well as theoretical analyses of chess games (Klein, 2022). Despite their successes, the trend of developing engines with handcrafted evaluation functions ended with the introduction of neural network-based chess engines such as AlphaZero (Silver et al., 2018). AlphaZero was trained through self-play to evaluate positions using a deep neural network instead of employing an evaluation function crafted by humans. Using this approach, AlphaZero was able to defeat the best traditional engine of its day by a wide margin.

Mimicking human chess players

As opposed to the development of superhuman chess engines, there has been comparatively less exploration of chess engines that predict human moves. McIlroy-Young et al. (2020)

In G. Boleda, I. Dautriche, M. Maldonado, C. Saldana and T. Torrent (Eds.), *Proceedings of the 48th Annual Conference of the Cognitive Science Society*

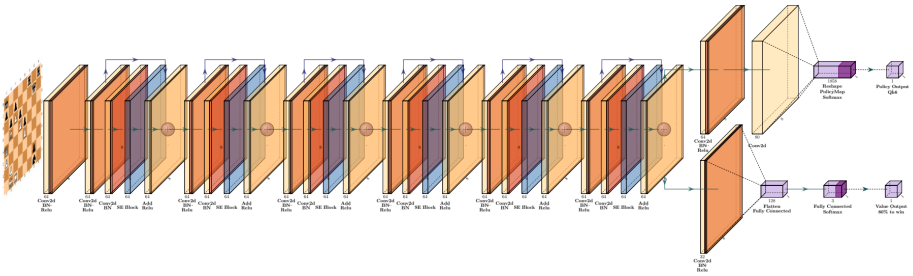


Figure 1: Maia model architecture from McIlroy-Young et al. (2020).

tackled this problem with the creation of a suite of 9 neural networks known as Maia. Maia models use a convolutional neural network architecture consisting of 6 blocks (Figure 1), and were trained with games from players corresponding to intermediate and advanced playing strengths as measured by their Elo ratings (Elo, 1978). The models were able to achieve a move-matching accuracy of 50-52% for the Elo range of games which they were trained on, with decreasing performance outside that range. Although these accuracies may appear low relative to other supervised learning tasks, the target label in this setting is inherently noisy: in a given position, different players of the same Elo (and even the same player across games) may choose different moves.

Interpreting concepts in chess networks

Interpretability methods have been applied to study chess neural networks that have achieved superhuman performance. Namely, McGrath et al. (2022) applied linear probes to AlphaZero to test whether its internal activations encode human-understandable chess concepts, such as king safety and piece mobility. A linear probe fits a linear regression to predict a board position's value on a given concept (e.g., its piece mobility) from a layer's activations for that board position. This would measure the extent to which the concept is linearly encoded in that layer's representation. McGrath et al. (2022) found that many human-understandable concepts are decodable from AlphaZero's activations, suggesting that AlphaZero makes use of familiar concepts to produce superhuman play. They also showed that the decodability of a concept varies systematically with the number of epochs that the network has been trained for and the depth of the layer of the probed activations, which indicates that how concepts are represented changes over learning and between layers.

Methods

To analyze the concepts encoded by neural networks trained to mimic human chess players, we applied the linear probing approach of McGrath et al. (2022) to the hidden activations produced by Maia models when they predict the moves made by humans for a given board position (McIlroy-Young et al., 2020). We also make use of the open-source chess engine Stockfish (`https://stockfishchess.org`), as well as a large set of chess games taken from the chess platform Lichess (`https://lichess.org`). In the following sections, we discuss how each of these components are used.

Data

We use the Chess Evaluations dataset from Kaggle (`https://www.kaggle.com/datasets`), which is a collection of positions sampled from games played on Lichess (`https://lichess.org`). Out of the dataset, we randomly sampled 65,534 positions for the training set and 8,192 for the test set to manage computational cost. Each position is represented by a Forsyth–Edwards Notation (FEN) string, which specifies the complete board state and other information needed to continue play.

Maia models

Maia consists of a suite of nine networks that each reproduce moves from human players of different Elo ratings. Each network is suffixed with an Elo that describes the class of players whose play it mimics. For instance, Maia 1100 is trained to predict the moves made by human players with an Elo of 1100-1199. We extract activations from these models to characterize which concepts are used and how they vary with skill level. Parts of the code used for the activation extraction are taken from Jenner et al. (2024).

Stockfish concepts

The goal of interpretability work on neural networks is to explain model behavior by using domain knowledge related to the task to specify how the model produces outputs (Zhang et al., 2021). Typically, this involves using domain knowledge to define task-relevant features and testing whether and where these features are encoded in the network's activations. In chess, a natural set of such features is provided by chess concepts, which can be defined as deterministic functions that map a board position to a real-valued score. Examples of chess concepts include the number of pawns possessed by white or the safety of black's king. These concepts serve as concise units of knowledge that chess players understand and that engines have historically used to evaluate board positions. These concepts are particularly useful for interpretability research because they extract game-relevant aspects of a position

Concept	MG	EG
Pawn	0.07	0.03
Knight	0.06	0.00
Bishop	−0.03	0.05
Rook	−0.18	−0.08
Queen	0.00	–
King Safety	−7.52	−0.04
Material	2.34	1.94
Imbalance	0.56	
Mobility	−0.71	−1.38
Threats	0.18	0.17
Passed Pawns	0.00	0.00
Space	−0.04	–
Total	−5.26	1.34
Final Eval	−5.18	

Figure 2: Example board position and concept scores. Stockfish 8 gives a final evaluation of -5.18 (black has a decisive advantage) for the given position (left). The corresponding concept score breakdown (right) shows that black has an unstoppable attack on the white king, with a mate-in-4 in this position.

that may be non-obvious from the raw board. For instance, the concept breakdown of the position in Figure 2 reveals that despite white's material advantage, the unsafe white king is much more relevant to the position and is responsible for Stockfish's evaluation of the position as winning for black.

Here, we make use of 24 chess concepts that are used as part of Stockfish 8's evaluation function, which correspond to those shown in Figure 2. These represent concepts that can be easily understood by chess players. Most concepts are defined separately for the middlegame (MG) and the endgame (EG), and thus suffixed by the phase of the game they represent. An important concept pair are the Total MG and Total EG concepts, which are computed from all concepts corresponding to the respective phase of the game. Amongst the 24 concepts, Final Evaluation is also included, which is a standalone concept that is computed as a weighted sum of the Total MG and Total EG values. McGrath et al. (2022) also used this set of concepts, allowing us to directly compare our results to theirs. A noteworthy point about the evaluation function is that it can only be used in positions where either side's king is not in check. As a result, I removed positions from the training and test set in which either side was in check, which decreased the training set size to 61,339 and the test set to 7,619. For positions where black is the player to move, the concept values are negated to align with the expectation that the neural networks are evaluating positions based on the side to move.

Linear probes

We follow the approach of McGrath et al. (2022) in using linear probes to test whether Maia's hidden activations encode interpretable chess concepts (Figure 3). Let $\mathbf{z}^d \in \mathbb{R}^{4096}$ denote the intermediate activation of layer d in a Maia model. Additionally, let $g_c(\mathbf{z}^0)$ be the score associated with the position representation as input to the neural network $\mathbf{z}^0$ for a given concept function g_c. A linear probe can be defined as $f_w(\mathbf{z}^d) = \mathbf{w}^\top \cdot [\mathbf{z}^d, 1]$, where 1 represents a bias term. Lastly, let Z be the training set. We then seek to train a linear probe to minimize the least-squares difference between the true concept value and the predicted concept value with L_1 regularization. More formally, the optimal linear probe can be defined as $f_{\mathbf{w}^\star}$, where $\mathbf{w}^\star$ is defined as:

$$\begin{aligned}\mathbf{w}^\star &= \underset{\mathbf{w}}{\operatorname{argmin}} \left(\sum_{\mathbf{z}^d \in Z} \| f_{\mathbf{w}}(\mathbf{z}^d) - g_c(\mathbf{z}^0) \|_2^2 + \lambda \|\mathbf{w}\|_1 \right) \\ &= \underset{\mathbf{w}}{\operatorname{argmin}} \left(\sum_{\mathbf{z}^d \in Z} \| \mathbf{w}^\top \cdot [\mathbf{z}^d, 1] - g_c(\mathbf{z}^0) \|_2^2 + \lambda \|\mathbf{w}\|_1 \right).\end{aligned}$$

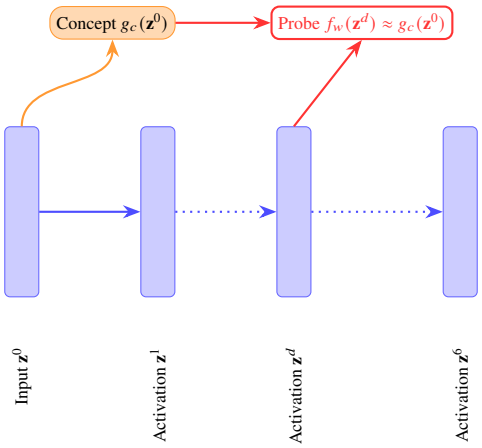


Figure 3: Probing activations in a Maia model. Graphic adapted from McGrath et al. (2022).

A positive scalar value of λ is chosen for regularization through the use of a validation set. Specifically, each linear probe is trained using multiple regularization constants over a portion of the training set. Each probe is then evaluated on an unseen validation set, and the regularization constant that yields the best performance is chosen. A final linear probe is then trained on the full training set. The training-validation split is 80-20 from the training set.

Results

We first report results on decoding concepts from Maia's representations. Then, we examine how decodability varies with the player-skill level the model emulates, and finally how it varies with model depth.

Stockfish concepts in Maia's latent space

We aim to assess whether Maia's activations encode human-understandable chess concepts, specifically those derived from Stockfish. To determine the extent to which these concepts were linearly decodable, we compute the coefficient of determination (R^2) of the linear probes as evaluated on the test set.

We found that all Stockfish concepts had positive R^2 scores with all intermediate activations of Maia models across all Elo scores (Table 1). This provides evidence that these concepts are all represented within the networks' activations to different degrees. Thus, despite the fact that these networks are trained only to predict human moves, their internal states still encode the same concepts used explicitly by Stockfish 8 and decodable from AlphaZero's activations. Given the ability of Maia models to predict human play, these results also suggest that the information required to predict human play in the Elo range of 1100-1900 involves these familiar concepts, motivating the hypothesis that players may also make moves using these features.

Effects of playing strength on concept presence

Given that we observed evidence that Stockfish concepts are decodable from Maia's activations, we next explored whether the decodability of concepts varies with the skill level that the model is trained to emulate. To understand this relationship while disentangling the effects of Elo from that of network depth, we performed a linear regression with both Elo and model depth as independent variables and R^2 score as the dependent variable:

$$R^2 = \beta_{\text{Elo}} \cdot \frac{\text{Elo}}{100} + \beta_{\text{layer}} \cdot \text{layer} + \beta_{\text{Elo\_layer\_intercept}}.$$

From this model, we performed a two-sided hypothesis test with a significance level of $\alpha = 0.05$ to determine whether there is a statistically significant relationship between Elo and R^2. This test revealed that 16 out of 24 concepts have a positive relationship between Elo and R^2 that is statistically significant (Table 2). We note that the impact of Elo was modest, with it only resulting in an increase of 0.3%-0.7% in R^2 score for every 100 Elo point. As an example, Figure 4 shows the relationship between the final evaluation concept and playing strength at model layer 6. Overall, these results illustrate that as models are trained to emulate human players of higher skill level, their internal representations become more predictive of Stockfish concepts. Treating the models as a proxy for actual human players, this pattern presents a suggestive hypothesis that better human players may rely more on these concepts when making a move compared to weaker players.

Statistic	Value
Total number of R^2 values	1296
Minimum	0.0803
Maximum	0.9484
Mean	0.5786
Median	0.6172

Table 1: Summary statistics of R^2 scores for all Maia models

Effects of model depth on concept presence in Maia

Model layer is also an important factor that can be associated with concept decodability, and the same linear regression model tests for this via β_{layer}. We found that 23 out of 24 concepts exhibited a statistically significant relationship between model layer and R^2 (Table 2). Out of those concepts, 4 showed increasing decodability with depth while the remaining 20 exhibited a negative relationship. Figures 5a and 5b illustrate these contrasting pattern, with Figure 5a showing a negative correlation between a simple concept (Pawn MG) and layer number and Figure 5b showing a positive correlation between a complex concept (Threats EG) and layer number.

We hypothesize that this difference in the direction of the relationship between model depth and concept decodability depends on concept complexity. More relational concepts may require integrating information across pieces and their interactions, and thus become increasingly decodable only in

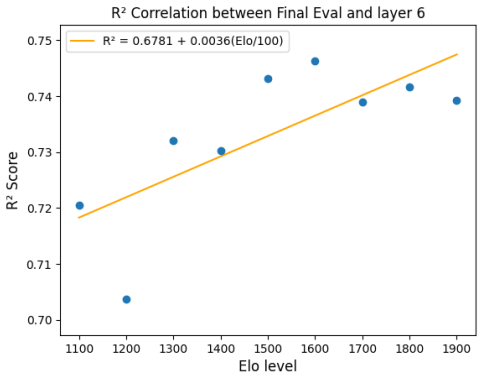


Figure 4: Relationship between the decodability of the final evaluation concept and playing strength as measured by Elo rating of the players the models are trained on in model layer 6.

Concept	β_{Elo}	β_{Elo} p-value	β_{layer}	β_{layer} p-value	$\beta_{\text{intercept}}$
Pawn MG	0.005	**0.000**	−0.032	**0.000**	0.332
Pawn EG	0.005	**0.000**	−0.033	**0.000**	0.318
Knight MG	0.004	**0.000**	−0.032	**0.000**	0.556
Knight EG	0.004	**0.002**	−0.027	**0.000**	0.552
Bishop MG	−0.000	0.718	−0.023	**0.000**	0.439
Bishop EG	0.004	**0.001**	−0.038	**0.000**	0.621
Rook MG	0.002	0.484	−0.008	**0.010**	0.595
Rook EG	0.000	0.917	−0.022	**0.000**	0.693
Queen MG	0.001	0.393	0.011	**0.000**	0.088
King Safety MG	0.002	0.098	0.014	**0.000**	0.340
King Safety EG	0.003	**0.001**	−0.004	**0.001**	0.177
Material MG	0.006	**0.000**	−0.015	**0.000**	0.837
Material EG	0.006	**0.000**	−0.014	**0.000**	0.843
Imbalance	0.004	**0.009**	−0.052	**0.000**	0.687
Mobility MG	0.002	0.111	−0.004	**0.023**	0.714
Mobility EG	0.001	0.625	−0.010	**0.000**	0.809
Threats MG	0.007	**0.013**	0.065	**0.000**	0.114
Threats EG	0.004	0.090	0.067	**0.000**	0.146
Passed Pawns MG	0.003	**0.031**	−0.012	**0.000**	0.667
Passed Pawns EG	0.003	0.052	−0.011	**0.000**	0.647
Space MG	0.003	**0.000**	−0.009	**0.000**	0.787
Total MG	0.003	**0.001**	−0.003	**0.015**	0.743
Total EG	0.004	**0.000**	−0.008	**0.000**	0.842
Final Eval	0.003	**0.001**	−0.001	0.205	0.710

Table 2: $\beta_{\text{intercept}}$, β_{Elo}, β_{layer}, and their corresponding p-values. Significant p-values ($\alpha = 0.05$) are shown in bold.

later layers. An example of this is endgame threats, which exhibit a significant positive relationship between layer number and concept presence. This concept involves pieces being undefended, attacking other pieces, or restricting the movement of other pieces. It is possible that later layers may increasingly encode these interactions between pieces after earlier layers have extracted more basic positional information. This pattern parallels findings in computer vision, where prior work has found that earlier layers represent low-level features such as edges and textures, and later layers represent more structured patterns and objects (Olah et al., 2020). On the other hand, simpler concepts are easier for the networks to identify from the initial input, and later layers move away from these representations. This behavior mirrors results from Lomasov et al. (2025) in that they also discover decreased concept alignment in later layers for a transformer-based chess engine. This suggests that networks make use of simpler Stockfish concepts in earlier layers and drift further away from these concepts in later representations.

To test this hypothesis, we define concept complexity as the number of leaf-level subconcepts in the computation tree of a given concept as specified in Stockfish's source code. The rationale behind this complexity measure is that it relies directly on subconcepts that are rooted in chess knowledge and have already been defined by the developers of Stockfish. The latter, in particular, required rigorous testing to determine logical decompositions for how certain concepts should be calculated. As such, this is an appropriate definition of concept complexity.

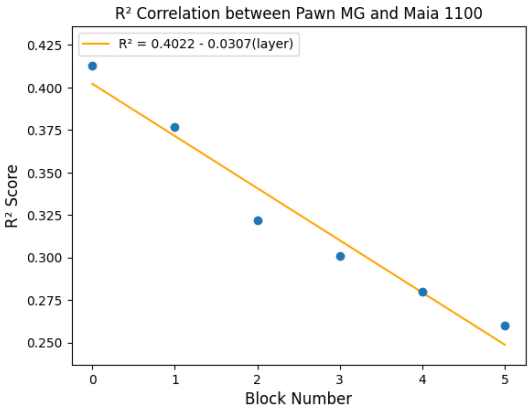
With this measure in hand, we fit a linear mixed-effects model with Elo, layer depth, concept complexity, and layer depth × concept complexity as independent variables and R^2 score as the dependent variable. We choose to use a mixed-effects model, with concept as the grouping variable, as groups of observations originate from the same concept and are not independent. Mathematically, the model can be written as:

$$R^2 = \beta_{\text{Elo}} \cdot \frac{\text{Elo}}{100} + \beta_{\text{layer}} \cdot \text{layer} + \beta_{\text{complexity}} \cdot \text{complexity} + \beta_{\text{layer}\times\text{complexity}} \cdot (\text{layer} \times \text{complexity}) + \beta_{\text{intercept}} + u_0 + u_1 \cdot \text{layer} + \varepsilon.$$

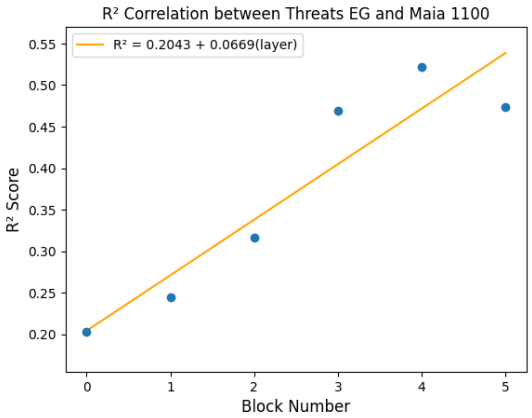
From this model, we performed a one-sided hypothesis test with a significance level of $\alpha = 0.05$ to determine if complexity causes a greater increase in R^2 across layer depth. The results show statistically significant evidence that $\beta_{\text{layer}} < 0$ ($p = 0.029$). This supports the conclusion that simpler concepts are well represented in earlier layers, and become less represented in later layers due to the models optimizing for move prediction. Additionally, this analysis suggest that more complex, relational concepts become easier to decode in later layers as they require more basic information to be extracted in earlier layers before they can be discovered in later layers.

Discussion

In this work, we characterized whether neural networks that emulate chess players encode human-understandable chess concepts in their internal representations, and whether the strength of these concept encodings change as a result of the skill level of the emulated players as well as network depth. We first chose a list of chess concepts that are concise and interpretable for human players. We then trained linear probes



(a) Relationship between the middle game pawn concept and layer number in Maia 1100. For this simple concept, there is a negative relationship between layer and R^2.



(b) Relationship between the end game threats concept and layer number in Maia 1100. For this complex concept, there is a positive relationship between layer and R^2.

Figure 5: Linear relationships between the layer of the Maia model and R^2 for two concepts with different degrees of complexity.

to measure these concepts in human-like models. In our analysis, we discovered that all concepts are decodable across models and layers. Additionally, we found that concept decodability increases with the Elo level of players the model is trained to emulate for many concepts, suggesting that these concepts become more relevant for predicting human moves as player skill increases. Finally, we investigated how concept decodability varies with network depth, speculating that the complexity of the concepts is a relevant factor in determining this relationship. We hypothesized that complex concepts see higher decodability in later layers compared to earlier layers, while simpler concepts experience a decrease in alignment for later layers as a result of concept drift. Our analysis supported the validity of this hypothesis, indicating that complexity does indeed play a role in how decodability changes across layers. Although our results concern the representations of behavior-cloned models rather than humans directly, they motivate the hypothesis that differences in chess expertise may reflect, in part, changes in how strongly and consistently these evaluative concepts are represented and integrated during move selection. This builds on previous work suggesting improved representations explain the development of chess expertise (Chase & Simon, 1973; Gobet, 1997).

Comparing our results to those obtained by McGrath et al. (2022), they also found that Stockfish concepts are decodable from AlphaZero's activations. Their analysis showed that concept decodability increased over training epochs. In other words, the stronger AlphaZero became, the more its activations became decodable to human-understandable concepts. Their results also showed that model layer affects the decodability of chess concepts. Overall, the direction of these effects aligned with what we observed in Maia. For example, McGrath et al. (2022) highlight a negative depth relationship for the middlegame pawn concept and a positive depth relationship for the endgame threats concept, which parallels the contrasting patterns that we identified.

Our analysis is complementary to a similar analysis performed on Maia-2, a transformer-based neural network that built upon the work of Maia (Tang et al., 2024). Their work also made use of the linear probing technique employed by McGrath et al. (2022), and discovered variability across concepts in whether their decodability were impacted by Elo. In our work, we made use of Maia despite the existence of Maia-2 as Maia allows for an analysis of how model depth impacts decodability. Additionally, the architecture of Maia is similar to AlphaZero in that they were both convolutional neural networks, which allows for a more direct comparison between our work and McGrath et al. (2022).

A limitation of this work is that we only probed the decodability of Stockfish evaluation concepts. While these concepts capture important aspects of a given chess position, they reflect a particular hand-designed feature set that is limited. In future work, we plan to include a broader range of chess concepts in order to better understand how complexity may impact the decodability of concepts in the internal representation of these networks. Some examples of simpler concepts include the number of pieces of a specific type present and the number of total pieces present, while examples of more complex concepts are piece coordination and prophylaxis, or restricting an opponent's mobility. Additionally, we aim to expand on this work by applying this technique on a larger collection of chess networks. Such an extension will allow us to determine whether some types of architectures produce representations that are more interpretable and easier to understand for human players. More broadly, this work is a step towards developing interpretable models for predicting human decisions in chess. With such a model in hand, chess would become an ideal setting for understanding how people plan and make decisions in complex environments.

Acknowledgments

This work was made possible by grants from the National Science Foundation (number IIS-2312373) and the NOMIS Foundation.

References

Campbell, M., Hoane, A., & Hsu, F.-h. (2002). Deep blue. *Artificial Intelligence*, *134*(1), 57–83. https://doi.org/https://doi.org/10.1016/S0004-3702(01)00129-1

Chase, W. G., & Simon, H. A. (1973). Perception in chess. *Cognitive psychology*, *4*(1), 55–81.

Elo, A. E. (1978). *The rating of chess players, past and present*. Arco Pub.

Gobet, F. (1997). A pattern-recognition theory of search in expert problem solving. *Thinking & Reasoning*, *3*(4), 291–313.

Gobet, F. (2018). *The psychology of chess*. Routledge.

Horton, J. J. (2023). Large language models as simulated economic agents: What can we learn from homo silicus? https://arxiv.org/abs/2301.07543

Jenner, E., Kapur, S., Georgiev, V., Allen, C., Emmons, S., & Russell, S. (2024). Evidence of learned look-ahead in a chess-playing neural network.

Ji-An, L., Benna, M. K., & Mattar, M. G. (2025). Discovering cognitive strategies with tiny recurrent neural networks. *Nature*, 1–9.

Klein, D. (2022). Neural networks for chess. https://arxiv.org/abs/2209.01506

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444.

Lomasov, S., Goldfeder, J., Erol, M. H., So, M., Yan, Y., Howard, A., Kutz, N., & Ziv, R. S. (2025). Exploring human-ai conceptual alignment through the prism of chess. https://arxiv.org/abs/2510.26025

McGrath, T., Kapishnikov, A., Tomašev, N., Pearce, A., Wattenberg, M., Hassabis, D., Kim, B., Paquet, U., & Kramnik, V. (2022). Acquisition of chess knowledge in alphazero. *Proceedings of the National Academy of Sciences*, *119*(47). https://doi.org/10.1073/pnas.2206625119

McIlroy-Young, R., Sen, S., Kleinberg, J., & Anderson, A. (2020). Aligning superhuman ai with human behavior: Chess as a model system, 1677–1687. https://doi.org/10.1145/3394486.3403219

Newell, A. (1955). The chess machine: An example of dealing with a complex task by adaptation. *Proceedings of the March 1-3, 1955, Western Joint Computer Conference*, 101–108.

Olah, C., Cammarata, N., Schubert, L., Goh, G., Petrov, M., & Carter, S. (2020). Zoom in: An introduction to circuits [https://distill.pub/2020/circuits/zoom-in]. *Distill*. https://doi.org/10.23915/distill.00024.001

Senel, L. K., Utlu, I., Yucesoy, V., Koc, A., & Cukur, T. (2018). Semantic structure and interpretability of word embeddings. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *26*(10), 1769–1779. https://doi.org/10.1109/taslp.2018.2837384

Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel, T., et al. (2018). A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, *362*(6419), 1140–1144.

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. *Advances in Neural Information Processing Systems 27*.

Tang, Z., Jiao, D., McIlroy-Young, R., Kleinberg, J., Sen, S., & Anderson, A. (2024). Maia-2: A unified model for human-ai alignment in chess. https://arxiv.org/abs/2409.20553

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems 30*.

Wang, A. Y., Kay, K., Naselaris, T., Tarr, M. J., & Wehbe, L. (2023). Natural language supervision with a large and diverse dataset builds better models of human high-level visual cortex. *bioRxiv*. https://doi.org/10.1101/2022.09.27.508760

Zhang, Y., Tino, P., Leonardis, A., & Tang, K. (2021). A survey on neural network interpretability. *IEEE Transactions on Emerging Topics in Computational Intelligence*, *5*(5), 726–742. https://doi.org/10.1109/tetci.2021.3100641