A Statistical Framework for Game-Based AI Evaluation

Anonymous Author(s)
Affiliation
Address
email

Abstract

We introduce a statistical framework for evaluating large language models (LLMs) in two-player games. The model separates premature endings, such as timeouts or repeated invalid moves, from the conditional outcome of win, draw, or loss. Both parts share a low-dimensional skill space for models and games, which lets us capture reliability (avoiding failures) and proficiency (winning valid games). Using the TextArena dataset (57 models, 30 games, about $38k$ matches including human players), we learn skills that can be used to compare similarity between LLMs' skill profiles, rank models, or predict performance in other tasks such as solving mathematical problems. In sum, our method turns arena outcomes into a structured and interpretable map of model reliability and capability.

1 Introduction

As static, single-turn benchmarks approach saturation, the field of large language models (LLMs) progresses increasingly depending on *interactive*, multi-turn evaluation that stresses instruction following, long context management, planning, and strategy. Game-based benchmarks and arenas provide such settings at scale. TextArena [13] offers a large, extensible collection of single-, two-, and multi-player text-based games, supports model-vs-model and model-vs-human play, and maintains a public, real-time leaderboard to track performance.

Standard summaries, such as raw win rates or separate ranks for each game type, still fall short. Treating each game in isolation ignores the fact that performance across games is often correlated: a model strong in chess may also show strengths in other strategic settings, though not in exactly the same way. Capturing these shared dimensions of skill requires going beyond independent per-game ranks can make the evaluation process more interpretable. Another issue is that invalid moves and timeouts are usually not given the needed attention. This wastes information, since such failures reveal important aspects of model behavior. For instance, producing outputs in the correct format, following instructions precisely, or avoiding hallucinated moves are all critical capabilities, and the frequency of invalid moves directly signals whether a model is reliable in these respects.

We propose a compact statistical model that breaks down game outcomes into two parts: whether the match ended prematurely (by timeout or by two invalid moves in a row) and, if it continued, whether it ended in a win/loss or draw. The two components of our model are linked through a shared low-dimensional skill space, much like in multidimensional Item Response Theory [22] model: models are characterized by latent skills, and each type of game outcome reflects a different combination of these skills. By modeling both valid and invalid outcomes, our framework captures a broader set of skills than traditional win-loss summaries. This includes not only the abilities needed to succeed in valid games but also the reliability-related skills required to avoid timeouts or invalid moves. Parameters are estimated via maximum likelihood and are identifiable up to a rotation, following conventions from factor models, *e.g.*, in [8]. Applied to TextArena data, the approach

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

reveals skill- and game-level patterns that extend the insights provided by current evaluation platforms and leaderboards.

2 Related work

Statistical modelling of games and matches. The Bradley–Terry (BT) model [2] is the standard framework for modeling competitive outcomes, with many extensions capturing richer structures. Examples include position bias or "home effects" [4], team-based models inferring individual skills [15], and dynamic or Bayesian formulations for longitudinal data [4, 23, 24]. To address intransitivity, the "Chest-and-Blade" framework uses attack and defense vectors [6, 7], later generalized for flexibility [11], with related ideas in competitor embeddings [5]. Beyond pairwise skills, extensions include Bayesian Mallows models for heterogeneous raters [10]. In the LLM setting, BT-type methods have been applied to Chatbot Arena data [9, 12].

Latent skills of LLMs. Performance correlations across benchmarks suggest LLMs share low-dimensional latent skills. Ilić [16] extract a general "g-factor" from the Open LLM Leaderboard [1] and GLUE [25], showing it correlates with model size. Using HELM [19], Burnell et al. [3] identify three interrelated factors that also scale with size, though without a formal scaling law; their analysis omits training set size and model family, limiting extrapolation. Kipnis et al. [18] apply unidimensional IRT to six Open LLM Leaderboard benchmarks, finding the primary factor ($\approx 80\%$ variance explained) aligns with overall leaderboard scores. Finally, Maia Polo et al. [20, 21] show that leveraging such latent skills can cut evaluation costs by up to 140×.

3 Methodology

We model head-to-head games between two LLMs, i_1 (first mover) and i_2 (second mover), that may end *prematurely* via either (i) a **timeout** or (ii) the player committing **two invalid moves in a row** ("two-strike" rule). If neither event occurs, the game proceeds to a valid conclusion with an outcome of win/draw/loss. Each LLM i has a skill vector $\theta_i \in \mathbb{R}^d$; game j has parameters detailed below. We write $\sigma(x) = 1/(1+e^{-x})$ for the sigmoid activation function.

3.1 Typed premature termination: timeouts and two-strike invalids

Let $Z_{i_1,i_2,j}$ denote the outcome random variable for game status, assuming it takes its values in

$$\big\{\text{“}i_1 \text{ timeout”},\ \text{“}i_2 \text{ timeout”},\ \text{“}i_1 \text{ two-strike”},\ \text{“}i_2 \text{ two-strike”}, \text{“valid game”}\big\}.$$

We use a multinomial logistic (softmax) model with "valid game" as the reference class. For each failure *type* $k \in \{\textsf{timeout}, \textsf{two-strike}\}$,

$$\log\left[\frac{\mathbf{P}(Z_{i_1,i_2,j} = \text{“}i_1, k\text{”})}{\mathbf{P}(Z_{i_1,i_2,j} = \text{“valid game”})}\right] = \delta_{j,k} + \lambda_{j,k} - \gamma_{j,k}^\top \theta_{i_1},$$
$$\log\left[\frac{\mathbf{P}(Z_{i_1,i_2,j} = \text{“}i_2, k\text{”})}{\mathbf{P}(Z_{i_1,i_2,j} = \text{“valid game”})}\right] = \lambda_{j,k} - \gamma_{j,k}^\top \theta_{i_2}.$$

Here, $\lambda_{j,k} \in \mathbb{R}$ is a type-specific base log-odds for game j, $\gamma_{j,k} \in \mathbb{R}^d$ links failure type k to skill (higher along $\gamma_{j,k}$ *reduces* that failure), $\delta_{j,k} \in \mathbb{R}$ captures a first-move *position bias* for type k. The position bias reflects the idea that players making more moves face a higher chance of ending the game prematurely.

3.2 Performance on valid games (win/draw/loss)

Let $Y_{i_1,i_2,j} \in \{\text{“invalid game”}, \text{“}i_1 \text{ wins”}, \text{“draw”}, \text{“}i_2 \text{ wins”}\}$. Consistency with the termination model is enforced by $\mathbf{P}(Y_{i_1,i_2,j} = \text{“invalid game”} \mid Z_{i_1,i_2,j} \neq \text{“valid game”}) = 1$. Conditional on validity, we use a paired-comparison model with a draw margin:

$$\mathbf{P}(Y_{i_1,i_2,j} = \text{“}i_1 \text{ wins”} \mid Z_{i_1,i_2,j} = \text{“valid game”}) = \sigma\big(\Delta_{i_1,i_2,j} - \beta_j\big),$$
$$\mathbf{P}(Y_{i_1,i_2,j} = \text{“}i_2 \text{ wins”} \mid Z_{i_1,i_2,j} = \text{“valid game”}) = \sigma\big(-\Delta_{i_1,i_2,j} - \beta_j\big),$$
$$\mathbf{P}(Y_{i_1,i_2,j} = \text{“draw”} \mid Z_{i_1,i_2,j} = \text{“valid game”}) = 1 - \sigma\big(\Delta_{i_1,i_2,j} - \beta_j\big) - \sigma\big(-\Delta_{i_1,i_2,j} - \beta_j\big),$$

with $\Delta_{i_1,i_2,j} = \alpha_j^\top(\theta_{i_1} - \theta_{i_2}) + \kappa_j$ with $\beta_j \geq 0$. Here $\alpha_j \in \mathbb{R}^d$ selects the skills governing valid-play performance on game j, κ_j is a position-bias term (advantage for moving first if $\kappa_j > 0$), and β_j is the draw margin (larger $\beta_j \Rightarrow$ more draws). Swapping i_1, i_2 flips the sign of Δ and exchanges the win probabilities. We fit the model using maximum-likelihood estimation with more details in Appendix A.

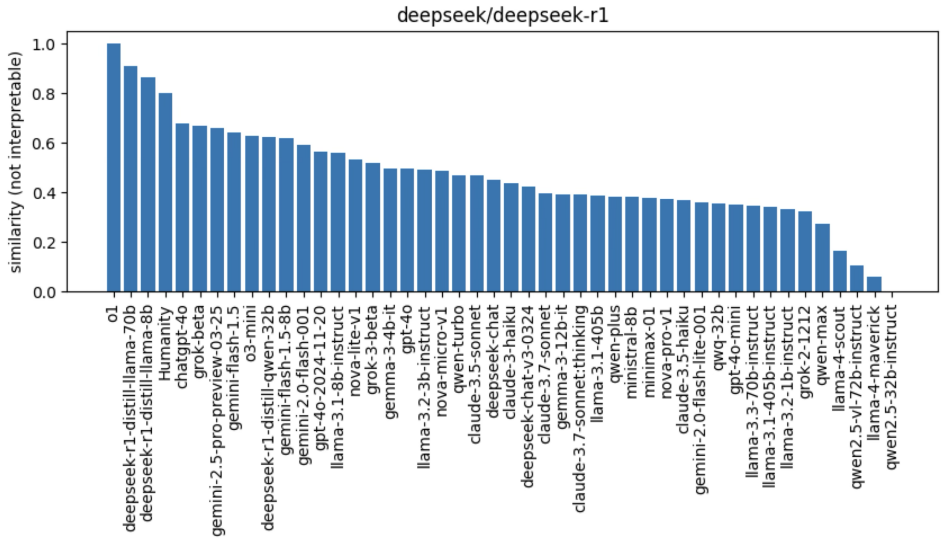

Figure 1: Cosine similarity between the skill profile of `deepseek-r1` and other LLMs, normalized so that the highest similarity is 1 and the lowest is 0. The plot highlights which models share the most similar latent skill structure with `deepseek-r1`.

3.3 Identifiability

The model is over-parameterized. To reduce the overparameterization, we fix the location and scale of the latent skills. A convenient convention is to *center and whiten* the skills across models, *i.e.*,

$$\frac{1}{n}\sum_i \theta_i = 0, \qquad \frac{1}{n}\sum_i \theta_i \theta_i^\top = I_d.$$

These constraints remove the global translation and scale indeterminacies. Even under centering and whitening, the likelihood is invariant to any common orthogonal rotation $R \in \mathbb{R}^{d\times d}$: replacing

$$\theta_i \leftarrow R^{-\top}\theta_i, \qquad \alpha_j \leftarrow R\alpha_j, \qquad \gamma_{j,k} \leftarrow R\gamma_{j,k}$$

leaves all probabilities (and thus the likelihood) unchanged because they depend only on inner products $\gamma_{j,k}^\top \theta_i$ and $\alpha_j^\top(\theta_{i_1} - \theta_{i_2})$. Consequently, parameters are identifiable only *up to a rotation* of the d-dimensional skill space. This type of rotational non-identifiability is standard in factor analysis and multidimensional IRT. In practice, one typically chooses a rotation after fitting the model to aid interpretation, for example, by applying a criterion such as *geomin* [17] to align skills with interpretable axes. The rotation does not change model fit or predictive performance but makes the latent dimensions easier to describe and compare across datasets.

4 Data analysis

In this section, we present preliminary results using the TextArena dataset, publicly available on HuggingFace[1] . The dataset includes 57 language models, ranging from general-purpose to frontier reasoning models, spanning 30 game types such as chess and other strategy games, with roughly 38k recorded matches. We filtered out games in which the number is valid games is less than 50 matches, ending up with 22 game modalities in total. A subset of the matches also involves human players. When fitting the model, we evaluated the validation loss on a small held-out subset of the data, which indicated that $d = 4$ is the optimal choice.

Comparing LLMs' skill profiles. After fitting our model, we can directly compare the latent skill profiles θ_i of different LLMs. A simple approach is to compute the cosine similarity between two vectors θ_i and $\theta_{i'}$, which measures the degree of alignment between their skill representations. In Figure 1, we plot these similarities, normalized so that the maximum value is 1 and the minimum is 0. The figure shows that `deepseek-r1` is most closely aligned with `deepseek-r1-distill-llama-70b`, `deepseek-r1-distill-llama-8b`, and OpenAI's `o1`, suggesting that these models have similar behavior in practical situations.

[1] https://huggingface.co/datasets/the-acorn-ai/textarena-player-game-traces

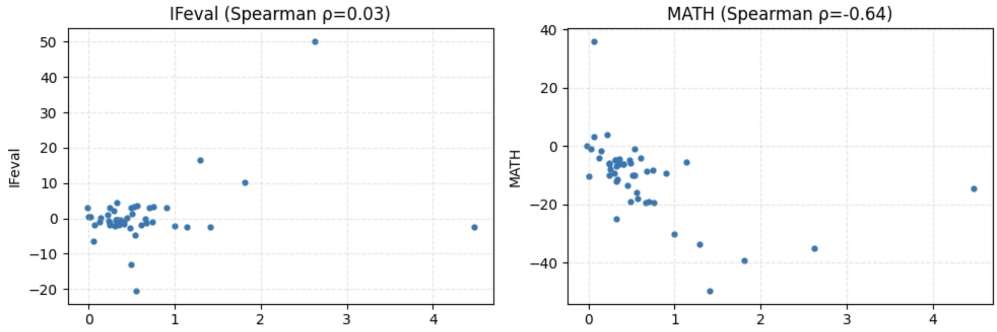

Figure 3: Pearson correlations between the TextArena complex instruction-following skill and skills estimated from one-dimensional IRT models on MATH and IFEval. The results show that the TextArena skill aligns more strongly with MATH than with IFEval, indicating that complex instruction-following in games captures reasoning abilities beyond those directly targeted by IFEval.

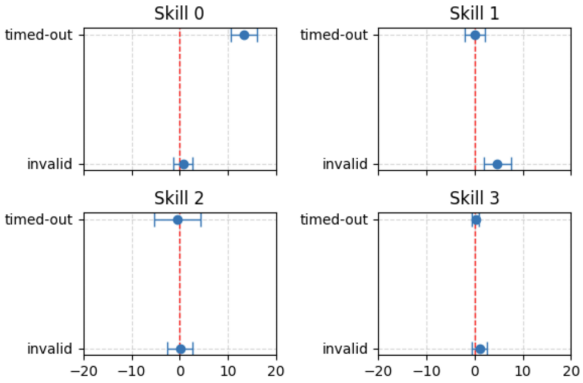

Figure 2: Rotated loadings ($\gamma_{j,k}$) averaged across games, with error bars showing standard deviations. After geomin rotation, "Skill 0" aligns with avoiding timeouts, while "Skill 1" aligns with avoiding invalid moves (following instructions).

Using rotations to find interpretable skills and rank models. As described in the methodology section, we can rotate the skill space using the *geomin* criterion to uncover more interpretable patterns. We perform the rotation of the model skills θ_i based on the loadings $\gamma_{j,k}$. Figure 2 shows the mean and standard deviation (across games j) of the rotated loadings for each skill dimension. A clear structure emerges: after rotation, "Skill 0" is strongly associated with avoiding timeouts, while "Skill 1" is linked to avoiding invalid moves, that is, following complex instructions correctly. In Figure 4, we use this interpretation to rank models by their ability to follow complex instructions. In Figure 5, we show the loadings α_j for all games. From that figure, we see that each skill can be more or less loaded in some games. For future steps, we plan to develop ways to interpret these loadings more insightfully.

Correlating TextArena skills with well-known benchmarks. One way to interpret what the latent skills represent in TextArena is to compare them with skills extracted from established benchmarks. To do this, we fit one-dimensional IRT models to MATH [14] (for mathematical problem solving) and IFEval [26] (for instruction following) and estimated skill parameters for the same 57 models. Figure 3 reports Pearson correlations between the TextArena complex instruction-following skill and the benchmark-derived skills. Since the model is invariant to translations, the sign of the correlation is not directly meaningful, as long as the relative alignment is interpreted consistently. We find that stronger complex instruction-following skills in TextArena correlate positively with higher MATH skills. Interestingly, the correlation is stronger with MATH than with IFEval, suggesting that what we label "complex instruction following" in games is more closely tied to mathematical reasoning than to the narrower instruction-following behaviors measured by IFEval.

References

[1] Edward Beeching, Clémentine Fourrier, Nathan Habib, Sheon Han, Nathan Lambert, Nazneen Rajani, Omar Sanseviero, Lewis Tunstall, and Thomas Wolf. Open llm leaderboard., 2023. URL https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard,2023.

[2] Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

[3] Ryan Burnell, Han Hao, Andrew RA Conway, and Jose Hernandez Orallo. Revealing the structure of language model capabilities. *arXiv preprint arXiv:2306.10062*, 2023.

[4] Manuela Cattelan, Cristiano Varin, and David Firth. Dynamic bradley–terry modelling of sports tournaments. *Journal of the Royal Statistical Society Series C: Applied Statistics*, 62(1): 135–150, 2013.

[5] David Causeur and François Husson. A 2-dimensional extension of the bradley–terry model for paired comparisons. *Journal of statistical planning and inference*, 135(2):245–259, 2005.

[6] Shuo Chen and Thorsten Joachims. Modeling intransitivity in matchup and comparison data. In *Proceedings of the ninth acm international conference on web search and data mining*, pages 227–236, 2016.

[7] Shuo Chen and Thorsten Joachims. Predicting matchups and preferences in context. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 775–784, 2016.

[8] Yunxiao Chen, Xiaoou Li, and Siliang Zhang. Joint maximum likelihood estimation for high-dimensional exploratory item factor analysis. *Psychometrika*, 84:124–146, 2019.

[9] Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E Gonzalez, et al. Chatbot arena: An open platform for evaluating llms by human preference. *arXiv preprint arXiv:2403.04132*, 2024.

[10] Marta Crispino, Elja Arjas, Valeria Vitelli, Natasha Barrett, and Arnoldo Frigessi. A bayesian mallows approach to nontransitive pair comparison data. *The Annals of Applied Statistics*, 13 (1):492–519, 2019.

[11] Jiuding Duan, Jiyi Li, Yukino Baba, and Hisashi Kashima. A generalized model for multidimensional intransitivity. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 840–852. Springer, 2017.

[12] Evan Frick, Connor Chen, Joseph Tennyson, Tianle Li, Wei-Lin Chiang, Anastasios N Angelopoulos, and Ion Stoica. Prompt-to-leaderboard. *arXiv preprint arXiv:2502.14855*, 2025.

[13] Leon Guertler, Bobby Cheng, Simon Yu, Bo Liu, Leshem Choshen, and Cheston Tan. Textarena. *arXiv preprint arXiv:2504.11442*, 2025.

[14] Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset, 2021. URL https://arxiv.org/abs/2103.03874.

[15] Tzu-Kuo Huang, Chih-Jen Lin, and Ruby Weng. A generalized bradley-terry model: From group competition to individual skill. *Advances in neural information processing systems*, 17, 2004.

[16] David Ilić. Unveiling the general intelligence factor in language models: A psychometric approach. *arXiv preprint arXiv:2310.11616*, 2023.

[17] Robert I Jennrich. Rotation. *The Wiley handbook of psychometric testing: A multidisciplinary reference on survey, scale and test development*, pages 279–304, 2018.

[18] Alex Kipnis, Konstantinos Voudouris, Luca M Schulze Buschoff, and Eric Schulz. metabench - a sparse benchmark to measure general ability in large language models. *arXiv preprint arXiv:2407.12844*, 2024.

[19] Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*, 2022.

[20] Felipe Maia Polo, Lucas Weber, Leshem Choshen, Yuekai Sun, Gongjun Xu, and Mikhail Yurochkin. tinybenchmarks: evaluating llms with fewer examples. *arXiv preprint arXiv:2402.14992*, 2024.

[21] Felipe Maia Polo, Ronald Xu, Lucas Weber, Mírian Silva, Onkar Bhardwaj, Leshem Choshen, Allysson Flavio Melo de Oliveira, Yuekai Sun, and Mikhail Yurochkin. Efficient multi-prompt evaluation of llms. *arXiv preprint arXiv:2405.17202*, 2024.

[22] M.D. Reckase. *Multidimensional Item Response Theory*. Springer New York, NY, 2009.

[23] Satoshi Usami. Individual differences multidimensional bradley-terry model using reversible jump markov chain monte carlo algorithm. *Behaviormetrika*, 37(2):135–155, 2010.

[24] Satoshi Usami. Bayesian longitudinal paired comparison model and its application to sports data using weighted likelihood bootstrap. *Communications in Statistics-Simulation and Computation*, 46(3):1974–1990, 2017.

[25] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*, 2018.

[26] Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models, 2023. URL https://arxiv.org/abs/2311.07911.

A Estimation via maximum likelihood

Let $\mathcal{D} = (i_1^{(m)}, i_2^{(m)}, j^{(m)}, Z^{(m)}, Y^{(m)})_{m=1}^{M}$ be the set of matches, where $Y^{(m)}$ is observed only when $Z^{(m)} =$ "valid game". Denote by $p_Z(i_1, i_2, j)$ the softmax probability of each termination label from the previous subsection, and by $p_{Y|\text{valid}}(i_1, i_2, j)$ the paired-comparison probabilities (win/draw/loss) from the valid-play model. The likelihood factorizes per match as

$$L(\Theta, \Gamma, \Lambda, \Delta) = \prod_{m=1}^{M} \left[\underbrace{p_Z\left(i_1^{(m)}, i_2^{(m)}, j^{(m)}\right)}_{\text{over } Z^{(m)}} \cdot \underbrace{p_{Y|\text{valid}}\left(i_1^{(m)}, i_2^{(m)}, j^{(m)}\right)^{\mathbf{1}\{Z^{(m)}=\text{valid}\}}}_{\text{only if valid}} \right],$$

where $\Theta = \theta_i$, $\Gamma = \{\gamma_{j,k}\}$, $\Lambda = \{\lambda_{j,k}, \delta_{j,k}, \kappa_j, \beta_j\}$, and $A = \{\alpha_j\}$. Equivalently, the log-likelihood is

$$\ell(\Theta, \Gamma, \Lambda, A) = \sum_{m=1}^{M} \left[\log p_Z\left(i_1^{(m)}, i_2^{(m)}, j^{(m)}\right) + \mathbf{1}\{Z^{(m)} = \text{valid}\} \log p_{Y|\text{valid}}\left(i_1^{(m)}, i_2^{(m)}, j^{(m)}\right) \right].$$

B Extra results

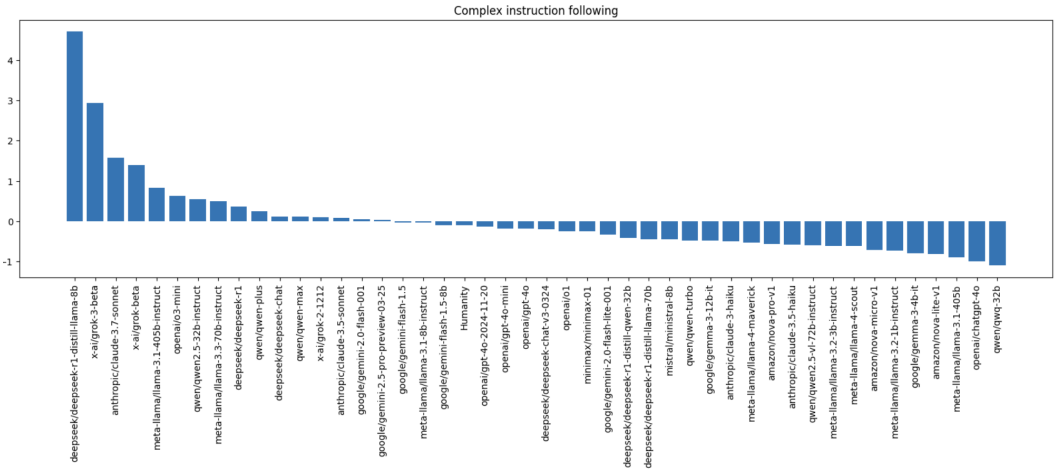

Figure 4: Model ranking by their instruction-following skill (Skill 1 after rotation). Higher values indicate a stronger ability to avoid invalid moves and follow complex instructions.

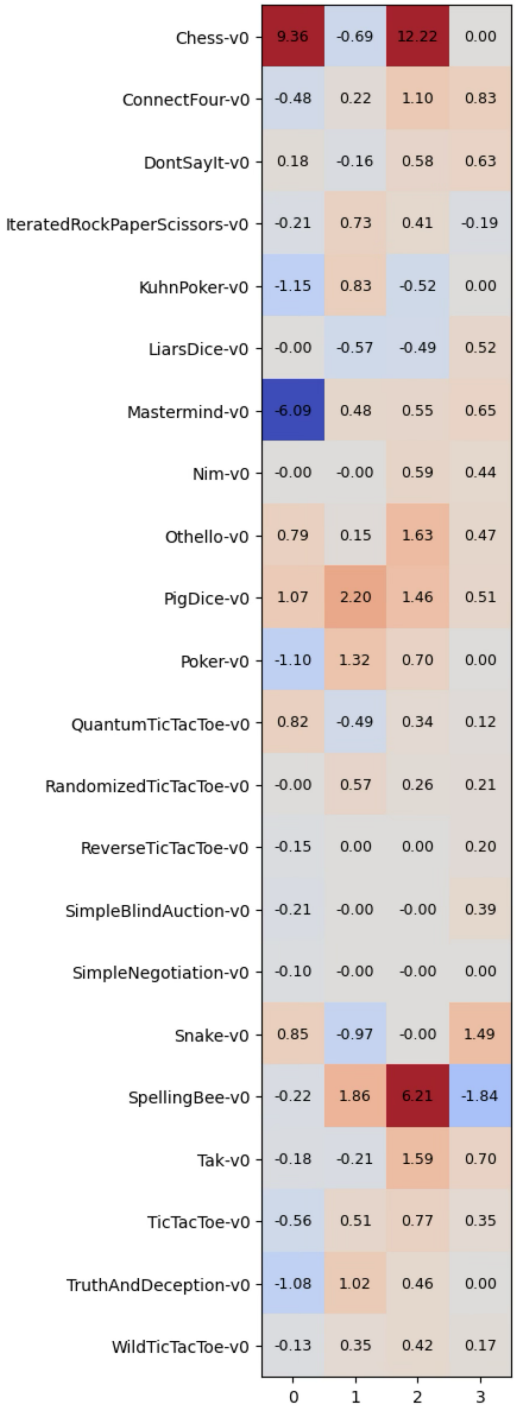

Figure 5: Loadings α_j