EVALUATING AND IMPROVING HUMAN BEHAVIOR MODELING IN CHESS

Anonymous authors
Paper under double-blind review

ABSTRACT

Human behavior models should reproduce not only likely actions, but also the stochastic structure and skill-dependent patterns of human decisions. We study rating-conditioned chess models and show that imitation policies can remain highly predictive while producing systematically mismatched move-quality distributions at higher ratings. To characterize this mismatch, we introduce three complementary diagnostics: mean highest-probability-density calibration error (mHPD-Err), which measures aggregate coverage calibration of high-probability move sets; move-class L1 error (Class L1), which measures differences in the frequencies of engine-derived move-quality classes; and move-class severity mean absolute error (Severity MAE), which measures differences in error severity within move-quality classes. We then propose an engine-informed information projection that minimally modifies the learned human policy to match population-level move-quality class frequencies and error severity. Across ratings 2100–2500, our method improves all three diagnostics relative to KL-regularized Monte Carlo tree search, while achieving comparable top-1 move matching accuracy and improving cross-entropy. At rating 2500, it reduces mHPD-Err from 5.10 to 1.73 percentage points, Class L1 from 1.89 to 0.43 percentage points, and Severity MAE from 1.027 to 0.010 centipawns, while also improving cross-entropy from 1.202 to 1.181 and top-1 move matching accuracy from 58.95% to 59.25% on the established ALLIE test set. These results demonstrate the importance of evaluating predictive accuracy, probability calibration, and skill-relative error structure as complementary aspects of human behavioral fidelity.

1 INTRODUCTION

Human behavior modeling aims to predict not only which action a person is most likely to take, but the distribution of actions expected from people of a given skill level. Chess is a useful setting because large-scale behavioral data, reliable player ratings, and strong external evaluators are available. Recent systems such as MAIA, MAIA-2, ALLIE, and CHESSFORMER have achieved increasingly strong performance in predicting human moves (McIlroy-Young et al., 2020; Tang et al., 2024; Zhang et al., 2025; Monroe et al., 2026).

One motivation for such models is *skill-conditioned position difficulty*: estimating how difficult a decision should be for a player of a given strength. Chess is widely used to study human cognition and performance (Strittmatter et al., 2020; Künn et al., 2022; 2023; Smerdon, 2022; Andolšek et al., 2025), typically using engine-derived measures such as centipawn loss (CPL). These quantify objective move quality, but not whether a move is unusually good or bad for the player making it. Reliable rating-conditioned policies could therefore provide a more appropriate behavioral reference for relative performance and position difficulty.

Before such models can be used for this purpose, we need to verify that their predicted policies reproduce human behavior. Related work typically evaluated the models using top-1 move matching accuracy (top-1 accuracy) and cross-entropy (CE). These provide only a partial view of a stochastic behavioral policy. We therefore propose three complementary metrics: mean highest-probability-density calibration error (mHPD-Err) for aggregate calibration of high-probability move sets, move-class L1 error (Class L1) for differences in the distribution of engine-derived move-quality classes, and move-class severity mean absolute error (Severity MAE) for differences in the severity of errors

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

within those classes. Together, these diagnostics reveal a systematic high-rating failure: the raw policy remains highly predictive and well calibrated under aggregate HPD coverage, yet becomes progressively too weak.

Previous work has addressed this problem using human-regularized Monte Carlo tree search (MCTS) (Jacob et al., 2022; Zhang et al., 2025). We find that KL-regularized search can recover human mean CPL, but does not fully recover the human error structure and worsens HPD coverage calibration. We instead seek the *smallest change to the learned human policy that reproduces population-level human move-quality statistics*. Using Stockfish annotations, we solve a KL-minimizing information projection that matches human move-quality frequencies and error severity while preserving the broad high-density structure of the original policy.

Our contributions are threefold. First, we develop a multidimensional evaluation of stochastic human chess policies beyond standard move accuracy and likelihood. Second, we identify a systematic move-quality deficit in high-rating imitation policies and show that matching mean CPL with MCTS does not necessarily recover human-like errors. Third, we introduce an engine-informed population projection that minimally modifies the learned human prior to correct this deficit while preserving better behavioral fidelity than the baseline. We open-source all code and data at our GitHub[1].

2 Related Work

Chess has long served as a testbed for artificial intelligence, from early work on game-tree search (Shannon, 1950; Turing, 2004) and high-performance systems such as Deep Blue (Campbell et al., 2002) to modern self-play agents such as AlphaZero (Silver et al., 2018). This tradition made chess a central domain for studying search and strategic decision making, but it largely optimized for stronger play rather than human-like play.

Human chess modeling shifts the objective from finding the strongest move to predicting the move distribution of a human player. MAIA introduced large-scale skill-specific move imitation (McIlroy-Young et al., 2020), followed by personalized models (McIlroy-Young et al., 2020) and style-specific models (McIlroy-Young et al., 2021; 2022; Tang et al., 2026). MAIA-2 unified multiple skill levels within a single rating-conditioned model (Tang et al., 2024), while more recent systems incorporate additional behavioral context such as time information, game history, pondering time, and resignation (Sekar & Jin, 2025; Zhang et al., 2025). CHESSFORMER further showed that transformer architectures provide a strong unified model for human move prediction and related chess tasks (Monroe et al., 2026); we use this architectural family as our human-policy prior.

Despite improvements in top-1 accuracy, prior work found that human-imitation policies fail to reproduce their target playing strength at higher ratings, motivating inference-time policy strengthening. Grill et al. (2020) interpret PUCT-style MCTS as regularized policy optimization, and Jacob et al. (2022) use KL-regularized search to improve a human policy while retaining proximity to the imitation prior. ALLIE similarly combines its learned human policy with KL-regularized MCTS and adaptive search (Zhang et al., 2025). These approaches motivate the MCTS baseline in our experiments.

Minimum-KL projection under expectation constraints is an established approach to incorporating distributional constraints into probabilistic models (Csiszár, 1975; Ganchev et al., 2010). Rather than estimating move values through repeated evaluations of the learned model, we use an external engine to annotate candidate moves with interpretable quality features and solve a population-level information projection. Our contribution lies in this behavioral formulation and its HPD-shell-preserving extension, which limits changes to the original policy's high-density structure.

3 Preliminaries

Let

$$\mathcal{D} = \{(s_i, r_i, a_i^{\mathrm{H}})\}_{i=1}^{N}$$

denote a dataset of chess positions s_i, active-player ratings r_i, and observed human moves a_i^{H}. For the legal action set $\mathcal{A}_i$, a rating-conditioned human model defines a stochastic policy $p_\theta(a \mid s_i, r_i)$.

[1]GitHub, link hidden for anonymity

Top-1 move matching accuracy. The standard move-matching metric is

$$\mathrm{Acc@1}(p_\theta) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}\left[a_i^{\mathrm{H}} = \arg\max_{a\in\mathcal{A}_i} p_\theta(a \mid s_i, r_i)\right]. \tag{1}$$

Top-1 accuracy evaluates only the most probable move and is invariant to any change in the remainder of the policy that preserves the argmax.

Cross-entropy. Related work also reports the negative log-likelihood of observed human moves,

$$\mathrm{CE}(q) = -\frac{1}{N}\sum_{i=1}^{N} \log q_i(a_i^{\mathrm{H}}). \tag{2}$$

CE is a proper scoring rule: in expectation it is minimized by the true conditional human distribution. Unlike top-1 accuracy, CE evaluates the probability assigned to the realized human action, but its absolute value depends on the intrinsic uncertainty of the population being modeled. Appendix B.1 discusses why CE alone does not characterize policy calibration.

Centipawn loss. For each legal move a, let $\ell_i(a) \geq 0$ denote its engine-estimated CPL relative to the engine-best move. The expected CPL of policy q_i is

$$\mathrm{CPL}(q) = \frac{1}{N'}\sum_{i\in\mathcal{I}_\ell}\sum_{a\in\mathcal{A}_i} q_i(a)\ell_i(a), \tag{3}$$

where $\mathcal{I}_\ell$ is the set of positions with usable CPL annotations and $N' = |\mathcal{I}_\ell|$. The corresponding human reference is the empirical mean $N'^{-1}\sum_{i\in\mathcal{I}_\ell}\ell_i(a_i^{\mathrm{H}})$. We use CPL as an engine-derived measure of move quality; it is not by itself a complete measure of human behavioral fidelity or game-level playing strength.

Temperature Scaling and Transcendence. Temperature scaling transforms a policy as $p_T(a \mid s, r) \propto p_\theta(a \mid s, r)^{1/T}$. The raw policy is recovered at $T = 1$; $T < 1$ *sharpens* the distribution toward high-probability moves, $T > 1$ *smooths* it, and $T \to 0$ approaches deterministic top-1 play. Prior work shows that sharpening can make a generative policy outperform the population used to train it, a phenomenon termed *transcendence* (Zhang et al., 2024). We reproduce this effect, and show that sharpening rapidly worsens HPD coverage calibration. Full strength and calibration sweeps are reported in Appendix B.4. We therefore use temperature scaling as a simple reference for changing policy strength rather than as a valid policy correction method.

4 METHODOLOGY

Our methodology has four components. We first train a rating-conditioned CHESSFORMER-style model. We then evaluate its stochastic behavior using complementary calibration and move-quality diagnostics. Next, we construct a KL-regularized MCTS baseline. Finally, we introduce an engine-informed population projection that corrects move-quality statistics while minimizing deviation from the human prior.

4.1 BASE HUMAN-MOVE MODEL

Our human-policy prior is a 23M-parameter no-history replication of CHESSFORMER (Monroe et al., 2026). It follows the original square-token representation, source–destination policy head, value head, rating conditioning, and geometric attention bias. Full data, model, training, replication details, and additional experiments are reported in Appendix A.

4.2 RATING-CONDITIONED MOVE-QUALITY CLASSIFICATION

We classify moves using an adaptation of the ChessLens implementation of Chess.com's Classification V2 expected-points approach (Chess.com, 2026; Abouchar, 2026). We evaluate legal moves with Stockfish 18 at depth 15. Deeper analysis is common for individual games, but exhaustive

evaluation over hundreds of thousands of positions makes search depth a major computational cost. Depth 15 provides a consistent move-quality signal, and all human and model policies within each comparison use identical engine settings. A controlled depth-sensitivity analysis (Table 33) shows that increasing depth from 15 to 18 changes absolute CPL but shifts every human and raw-policy class frequency by less than 0.5 percentage points. Engine settings and computational cost are reported in Appendix F.

The six quality classes are not fixed CPL intervals. For a candidate with engine score s from the side-to-move perspective, we approximate expected points as

$$\mathrm{EP}(s, r) = \sigma(0.00368\, g(r) s)\,, \qquad g(r) = 0.85 + 0.70\, \sigma(0.010(r - 1500))\,, \tag{4}$$

where r is the player's rating and σ is the logistic sigmoid. Mate scores map directly to expected points 1 or 0. The raw loss of move a is the difference between the expected points of the engine-best move and those of a. Following ChessLens, sufficiently large losses in draw-heavy positions receive an additional draw-sensitivity multiplier of at most 1.30.

The resulting effective expected-points loss is mapped to the published boundaries: 0 for *best*, $(0, 0.02]$ for *excellent*, $(0.02, 0.05]$ for *good*, $(0.05, 0.10)$ for *inaccuracy*, $[0.10, 0.20)$ for *mistake*, and losses of at least 0.20 for *blunder*. We omit brilliant, great, miss, and other contextual labels whose definitions require game history or additional tactical context. CPL is retained separately as a rating-independent engine-score loss.

Because the expected-points transformation becomes more sensitive to engine-score changes as rating increases, the same CPL can correspond to different classes for a novice and an expert. We therefore classify each cohort at its corresponding rating: the classes capture skill-relative error type, while CPL measures objective engine loss.

4.3 Behavioral Evaluation Criteria

No single statistic identifies the full human move distribution from a dataset containing one observed action per position. We therefore evaluate policies along complementary dimensions: predictive likelihood, aggregate coverage calibration, objective move quality, move-quality composition, and conditional error severity.

HPD coverage calibration. For a policy q_i and probability level α, let $S_{i,\alpha}(q)$ be the smallest prefix of legal moves, ordered by decreasing $q_i(a)$, whose cumulative mass reaches α. Its realized mass is

$$m_{i,\alpha}(q) = \sum_{a \in S_{i,\alpha}(q)} q_i(a). \tag{5}$$

Because actions are discrete, $m_{i,\alpha}$ generally exceeds α. We therefore compare empirical human coverage against the *realized*, rather than nominal, probability mass:

$$\text{HPD-Err}_\alpha(q) = \left| \frac{1}{N} \sum_i \mathbb{I}\big[a_i^{\mathrm{H}} \in S_{i,\alpha}(q)\big] - \frac{1}{N} \sum_i m_{i,\alpha}(q) \right|. \tag{6}$$

We average over $\mathcal{H} = \{0.25, 0.50, 0.60, 0.70, 0.80, 0.90\}$:

$$\text{mHPD-Err}(q) = \frac{1}{|\mathcal{H}|} \sum_{\alpha \in \mathcal{H}} \text{HPD-Err}_\alpha(q). \tag{7}$$

We evaluate multiple HPD levels to capture calibration across both the high-probability moves and lower-ranked alternatives. Appendix C reports realized masses, set sizes, sensitivity to the evaluation levels, and limitations of this aggregate diagnostic. mHPD-Err is reported in percentage points (pp).

Move-class composition. Let $P_q(c)$ denote the average probability assigned by policy q to class c, and let $P_{\mathrm{H}}(c)$ be its empirical human frequency. We define

$$\text{ClassL1}(q) = \sum_{c \in \mathcal{C}} \left| P_q(c) - P_{\mathrm{H}}(c) \right|. \tag{8}$$

The corresponding total-variation distance is one half of this value. Class L1 tests whether the policy produces the correct mixture of best moves, inaccuracies, mistakes, and blunders, rather than only the correct average CPL. The metric is expressed in pp.

Error severity. Matching class frequencies is still insufficient if, for example, the policy produces blunders at the correct frequency but with larger CPL than human blunders. Let $\mu_q(c)$ and $\mu_{\mathrm{H}}(c)$ denote policy and human mean CPL conditional on class c. We report

$$\mathrm{SeverityMAE}(q) = \sum_{c \in \mathcal{C}} P_{\mathrm{H}}(c) \left| \mu_q(c) - \mu_{\mathrm{H}}(c) \right|. \tag{9}$$

This metric is expressed in centipawns (cp).

4.4 KL-Regularized MCTS Baseline

Our search baseline follows the KL-regularized MCTS formulation used for human-policy correction by Jacob et al. (2022), building on the interpretation of PUCT as regularized policy optimization from Grill et al. (2020). At a root position s, the imitation model provides both the move prior $p_\theta(a \mid s, r)$ and value estimates used by MCTS.

Given searched root values $\widehat{Q}(s, a)$, the corrected policy solves

$$q^\star = \arg \max_{q \in \Delta(\mathcal{A}(s))} \sum_a q(a) \widehat{Q}(s, a) - \lambda D_{\mathrm{KL}}(p_\theta(\cdot \mid s, r) \,\|\, q). \tag{10}$$

For budget B, including the root evaluation, the regularization coefficient is induced by the PUCT schedule,

$$\lambda_B = \frac{c_{\mathrm{puct}} \sqrt{B - 1}}{k + B - 1}, \tag{11}$$

with $k = 0$ in the experiments.

Because stronger search can trivially lower CPL by moving farther from the human policy, c_{puct} is calibrated so that MCTS approximately matches the human mean CPL for each rating cohort. This gives a CPL-matched behavioral baseline: once mean engine-estimated move quality is held fixed, we can ask whether the resulting stochastic policy reproduces the remaining properties of human behavior.

4.5 Engine-Informed Population Projection

Matching mean CPL does not specify *how* a policy makes errors. We therefore construct the corrected policy directly from the mismatch observed in the move-quality distribution. Let $p_i(a)$ denote the human-model prior over legal move a in position i, and let $f_i(a)$ contain engine-derived features of that move. We seek the smallest average KL change to the human policy whose population moments match those of observed humans:

$$\min_{\{q_i\}} \quad \frac{1}{N} \sum_{i=1}^{N} D_{\mathrm{KL}}(q_i \,\|\, p_i), \tag{12}$$

$$\text{subject to} \quad \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_{a \sim q_i}[f_i(a)] = t. \tag{13}$$

The target vector t is estimated from observed human moves in a calibration population. It contains aggregate population moments rather than a separate target for each position.

The solution is a shared exponential tilt,

$$q_i(a; \theta) = \frac{p_i(a) \exp\big(\theta^\top f_i(a)\big)}{\sum_{b \in \mathcal{A}_i} p_i(b) \exp(\theta^\top f_i(b))}. \tag{14}$$

Thus, the engine does not replace the human prior. It only reweights the relative probabilities already supplied by the human model, using the same low-dimensional parameter vector θ across every position in a rating cohort.

With classes ordered as best, excellent, good, inaccuracy, mistake, and blunder, the feature vector is

$$\begin{aligned} f_i(a) = \big[&\mathbb{I}[c_i(a) = \text{best}], \ldots, \mathbb{I}[c_i(a) = \text{mistake}], \\ &\widetilde{\ell}_i(a)\mathbb{I}[c_i(a) = \text{excellent}], \ldots, \widetilde{\ell}_i(a)\mathbb{I}[c_i(a) = \text{blunder}]\big], \end{aligned} \tag{15}$$

where $\widetilde{\ell}_i(a) = \ell_i(a)/100$. The blunder-class indicator is the reference category and is omitted because class probabilities sum to one; best-move severity is omitted because best moves have zero CPL. The resulting model therefore has ten fitted coefficients: five controlling class mass and five controlling non-best error severity.

The corresponding human targets are the population frequencies of best through mistake and, for excellent through blunder, the population means of $\widetilde{\ell}_i(a)\mathbb{I}[c_i(a) = c]$. Matching these moments jointly controls class frequencies, class-conditional mean severity, and therefore overall expected CPL. This distinguishes policies with identical mean CPL but different error frequencies and severities.

We solve the convex dual using damped Newton updates and backtracking line search, with a numerical ℓ_2 term of 10^{-10}; feasibility checks and optimization residuals are reported in Appendix D.6. If prior probability mass lacks a usable engine annotation, that mass is held fixed and the projection is applied conditionally to the annotated support.

To prevent a held-out human action from influencing its own corrected policy, we use fixed five-fold cross-validation. Human target moments and θ are fitted on four folds and applied unchanged to the fifth; reported results concatenate predictions from all held-out folds.

4.6 HPD-Preserving Projection

The unconstrained class+severity projection matches human move-quality moments but can transfer probability between the head and tail of the raw policy, increasing mHPD-Err. We therefore add a per-position constraint that preserves the raw policy's broad high-density structure.

For each position, let the head shell be the smallest raw-policy prefix $S_{i,0.80}(p)$ whose mass reaches 0.80, and let the second shell contain the remaining moves. If M_{is} is the original probability mass of shell s, we preserve this mass exactly and apply the shared engine tilt only *within* each shell:

$$q_i(a;\theta) = M_{is}\frac{p_i(a)\exp\big(\theta^\top f_i(a)\big)}{\sum_{b\in s} p_i(b)\exp(\theta^\top f_i(b))}, \qquad a \in s. \tag{16}$$

This is equivalent to minimizing the full-policy KL objective under both the population class+severity constraints and fixed per-position shell masses. The engine may therefore redistribute probability among moves within the raw high-density region and within its tail, but cannot move probability directly between the two. Alternative shell definitions and the effect of HPD preservation are evaluated in Appendix D.4.

5 Results

We first characterize the raw policy's move-quality mismatch, then examine whether CPL-matched MCTS or our engine-informed projection recovers human-like behavior. Finally, we evaluate the projected agents' game-level rating structure. All experiments use the 23M model from Section 4.1.

5.1 Characterizing the High-Rating Move-Quality Deficit

Prior work has observed that human-imitation chess policies can fail to reproduce the intended playing strength at higher ratings (Jacob et al., 2022; Zhang et al., 2025). Our engine-based analysis characterizes how this failure arises. As rating increases, the raw policy assigns progressively too little probability to engine-best moves and too much to mistakes and blunders. At lower ratings the mismatch is comparatively modest and partly reversed, in a direction that can be addressed by smoothing with $T > 1$. We therefore focus subsequent correction experiments on the high-rating regime, where the deficit grows rapidly and naive sharpening would substantially distort the predicted distribution. The complete rating sweep and move-class distributions are reported in Appendix B.2.

This class mismatch is notable because the raw policy still has very low mHPD-Err (Table 2). Small probability errors across plausible moves—particularly low-probability mistakes and blunders—can have little effect on aggregate HPD coverage while materially affecting expected move quality when the policy is sampled repeatedly. Estimating such probabilities is inherently difficult when

each position contributes only one observed move. This motivates correcting move-quality class frequencies and error severity rather than only matching mean CPL.

5.2 Matching Mean CPL Does Not Recover Human-Like Error Structure

We tune c_{puct} separately for each rating so that MCTS approximately matches human mean CPL. We use $B = 50$, matching the primary search budgets in Jacob et al. (2022) and ALLIE (Zhang et al., 2025). Appendix D.1 additionally evaluates budgets up to $B = 200$, the maximum used by ALLIE's adaptive search, and shows diminishing improvements in behavioral fidelity at matched mean CPL.

Matching mean CPL does not guarantee human-like errors. At rating 2500 (Table 1), MCTS assigns too much probability to best moves and too little to mistakes and blunders despite matching human mean CPL. The same pattern appears at lower ratings, but differences are less apparent, since the raw-policy quality deficit is smaller (Appendix D.2).

Table 1: Move-class distribution at rating 2500 in the ALLIE set. MCTS and Projection policies both approximately match human mean CPL, but differ in the composition of that loss.

	Human	Raw	MCTS	Projection
Best (%)	46.63	41.99	47.49	46.42
Excellent (%)	19.14	18.57	19.23	19.14
Good (%)	14.35	14.39	13.89	14.39
Inaccuracy (%)	8.66	9.68	8.49	8.69
Mistake (%)	7.07	8.78	6.88	7.11
Blunder (%)	4.16	6.59	4.03	4.25
CPL	33.93	46.80	34.10	33.93

The apparent contradiction is resolved by considering error severity. MCTS blunders average 273.52 CPL, compared with 264.15 CPL for human blunders. Fewer but larger errors can therefore offset an excess of best moves and produce the same mean CPL. This motivates the class+severity constraints used by our engine projection.

5.3 Engine Projection Recovers Move Quality with Better Behavioral Fidelity

Table 2 compares the raw policy, CPL-matched MCTS, and our five-fold out-of-fold 80%-shell projection. The raw policy exhibits a growing move-quality deficit despite low mHPD-Err, while MCTS corrects mean CPL at the cost of worse HPD calibration.

The engine projection also matches human move quality while retaining better HPD calibration. Its low Class L1 and Severity MAE are expected because these quantities correspond directly to fitted population moments. Importantly, CE also improves over both MCTS and the raw policy throughout the evaluated high-rating range. mHPD-Err remains below that of MCTS, although higher than for the uncorrected raw policy. Top-1 accuracy changes little across methods. This is precisely the setting in which a multidimensional evaluation is useful: policies with nearly identical top-1 accuracy can differ materially in expected move quality, error composition, likelihood, and probability calibration.

To isolate the benefit of the richer behavioral targets from access to Stockfish itself, we also fit a one-parameter CPL-only projection using the same human prior, engine annotations, folds, optimizer, and 80% shell. Although it matches human mean CPL, it retains larger Class L1 (2.37 pp) and Severity MAE (1.164 cp) than the full projection (0.49 pp and 0.031 cp, respectively). Class-only and severity-only ablations further show that the two feature families address complementary aspects of the error distribution, while replacing the learned human prior with a uniform prior severely degrades predictive performance (Appendix D.3).

To test generalization beyond the ALLIE evaluation population, we apply the correction to a much larger post-training Lichess cohort. At rating 2400 (Table 3), the frozen ALLIE-fitted projection generalizes without refitting: it improves top-1 accuracy and CE and reduces the raw move-quality, class, and severity mismatches. It does not remain exactly aligned with the broader population

Table 2: Full-set comparison of the raw policy, CPL-matched KL-regularized MCTS ($B = 50$), and the five-fold out-of-fold engine class+severity projection with exact preservation of the raw 80% HPD-shell mass. MCTS uses $c_{\text{puct}} = \{4.0, 2.7, 2.2, 1.8, 1.2\}$ for increasing ratings respectively. ΔCPL is relative to the human mean; values closer to zero are better. Across rating cohorts, position-level 95% CI half-widths are approximately 0.41–0.55 pp for top-1 accuracy, 0.010–0.015 for CE, 0.22–0.40 pp for mHPD-Err, 0.74–1.12 pp for Class L1, and at most 1.30 cp for Severity MAE.

Rating	N	Method	Top-1 acc. (%) ↑	CE ↓	ΔCPL	mHPD-Err ↓	Class L1 ↓	Sev. MAE ↓
2100	50,936	Raw model	57.05	1.275	+6.1	0.38	3.59	2.231
		MCTS	57.29	1.274	−0.3	1.82	0.94	0.640
		Projection	57.29	1.273	−0.1	0.70	0.52	0.076
2200	52,970	Raw model	57.57	1.256	+7.9	0.31	4.66	2.638
		MCTS	57.70	1.256	−0.3	2.45	1.74	0.564
		Projection	57.69	1.252	0.0	0.79	0.44	0.032
2300	54,737	Raw model	58.45	1.227	+9.1	0.18	6.57	2.413
		MCTS	58.92	1.226	−0.2	2.73	1.30	0.702
		Projection	58.72	1.222	0.0	0.85	0.44	0.030
2400	48,223	Raw model	58.86	1.203	+10.3	0.19	7.67	2.732
		MCTS	59.19	1.202	−0.2	3.28	1.53	0.838
		Projection	59.22	1.195	0.0	0.98	0.49	0.031
2500	30,465	Raw model	58.66	1.194	+12.9	0.54	10.41	3.101
		MCTS	58.95	1.202	+0.2	5.10	1.89	1.027
		Projection	59.25	1.181	0.0	1.73	0.43	0.010

because the target moments themselves differ across populations; for example, human mean CPL is 35.80 in broader Lichess versus 39.29 in ALLIE set. Refitting on the broader population removes this residual mismatch while retaining the predictive improvement, indicating that the correction generalizes but its target moments are population dependent. Full transfer and broader projection results across all ratings are reported in Appendices E.2 and E.3 respectively.

Table 3: Population transfer at rating 2400. Policies are evaluated on 410,096 positions from the broader Lichess cohort. The ALLIE projection uses parameters fitted on the full ALLIE rating-2400 cohort; the BL projection is fitted by five-fold out-of-fold evaluation on the broader Lichess cohort. ΔCPL is relative to the broader-Lichess human mean. mHPD-Err and Class L1 are reported in percentage points.

Policy	Top-1 (%)	CE	ΔCPL	mHPD-Err	Class L1	Sev. MAE
Raw policy	58.47	1.220	+13.46	0.47	11.53	3.130
ALLIE projection	59.17	1.203	+3.04	0.93	4.58	1.547
BL projection	59.21	1.204	+0.01	1.67	0.42	0.014

5.4 GAME-LEVEL VALIDATION OF RATING STRUCTURE

We have deliberately referred to the corrected policies as *CPL-matched* rather than strength-matched: matching CPL does not establish that an agent realizes the intended playing strength in games, because CPL depends on the difficulty of the positions (Appendix E.1). We therefore run 1,000-game matches between projected agents at rating pairs from 2100 to 2500. For game play, we refit the 80%-shell projection on the full ALLIE cohort and use adaptive 99.9% engine coverage (Appendix D.7). We compare match scores against empirical Lichess blitz results rather than classical Elo expectations (Elo, 1978), since Lichess uses Glicko-2 (Glickman, 2001) and outcomes also depend on playing platform and playing conditions. Table 4 shows that the lower-rated projected agent scores above the broader human reference in all ten pairs, by 5.26 pp on average. The projected policies therefore exhibit a more compressed rating structure.

However, refitting the projection on the much larger broader-Lichess cohorts does not recover the empirical game-level rating gaps: the lower-rated agents still score 5.77 pp above the empirical

Table 4: Game-level comparison with a broader empirical sample of Lichess blitz games. Each projected-policy match contains 1,000 games, with 95% CIs of approximately ± 2.8 pp. The empirical reference contains approximately 2,500 games per rating pair, with 95% CIs of approximately ± 1.5–2.0 percentage points. Scores are from the perspective of the lower-rated player.

Rating pair	Broader Lichess (%)	Projected (%)	Δ (pp)
2100–2200	39.90	41.70	+1.80
2100–2300	28.90	35.20	+6.30
2100–2400	21.30	28.50	+7.20
2100–2500	15.50	21.10	+5.60
2200–2300	38.80	43.20	+4.40
2200–2400	28.60	37.15	+8.55
2200–2500	21.50	28.05	+6.55
2300–2400	39.60	42.80	+3.20
2300–2500	29.10	35.80	+6.70
2400–2500	37.10	39.40	+2.30

reference on average (Appendix E.4), indicating that calibration-population shift is not the only limitation. A phase-specific analysis identifies one plausible additional factor: the 30-second clock filter used in both model training and projection calibration removes approximately half of available endgame positions, while agent errors increase sharply in practical endgames (Appendix E.5). A small deployment against human opponents shows the same pattern: the agent has lower mean CPL than its opponents in the opening and middlegame but higher mean CPL in the endgame (Appendix E.6). This means that training and calibration systematically underrepresent many endgame positions that contribute to full-game performance, potentially limiting the agents' realized playing strength.

6 Conclusion

We show that rating-conditioned human chess policies can remain highly predictive and well calibrated under aggregate HPD coverage while producing systematically mismatched move-quality distributions at high ratings. CPL-matched MCTS corrects mean move quality but does not recover human error composition or severity, and worsens HPD calibration. Our engine-informed information projection instead makes a minimum-KL population correction that matches human move-quality statistics while retaining much better HPD calibration than MCTS. On held-out positions, the corrected policies also improve human-move likelihood over both MCTS and the raw policy.

More broadly, behavioral fidelity is multidimensional. CE measures predictive likelihood, mHPD-Err measures aggregate coverage calibration, and the class and severity metrics characterize the skill-relative structure of errors. A model can perform well on one of these dimensions while failing on another. This distinction matters for applications such as skill-conditioned position difficulty and performance assessment, where the relevant question is not only whether a move is objectively good, but whether it is expected from a player of a given strength.

Several limitations remain. The main experiments cover ratings 2100–2500 and one CHESSFORMER-style policy family. Our larger balanced rating-pair analysis shows that move-quality targets depend on the population of positions encountered, including opponent strength, and an ALLIE-fitted projection therefore does not transfer perfectly to broader Lichess cohorts. More importantly, even projections recalibrated on the broader population do not fully reproduce empirical game-level rating gaps. One plausible contributor is the 30-second clock filter used in both training and calibration, which disproportionately removes endgames and leaves practical late-game behavior underrepresented. Future models should therefore account for opponent strength, game phase, and time context rather than treating rating as the only source of behavioral heterogeneity. Finally, internal agent tournaments cannot establish absolute human-equivalent rating; confirming this ultimately requires large-scale play against human players.

AI USE STATEMENT

Generative AI tools were used for language editing, code assistance, feedback on research methodology and experimental design, and assistance with the presentation of results. All AI-assisted suggestions, implementations, numerical results, analyses, and scientific claims were independently reviewed and verified by the authors.

ETHICS STATEMENT

This work studies models trained to imitate human chess behavior using public game data. The experiments operate on chess positions, moves, ratings, and game metadata and do not require personally identifying information. The small human-opponent deployment used only public game records from games played through a clearly identified Lichess bot account; no private user information was collected. Human-like chess agents have beneficial applications in tutoring, behavioral research, and performance assessment, but could also be misused to assist cheating while appearing less engine-like than conventional chess software. We therefore view the study of such models, including their characteristic error distributions, as relevant both to beneficial human-centered applications and to future work on robust cheating detection.

REPRODUCIBILITY STATEMENT

We provide the information needed to reproduce the main experiments throughout the paper and supplementary material. Section 4 specifies the behavioral metrics, MCTS baseline, engine-informed population projection, HPD-preserving constraint, and cross-validation procedure. Appendix A describes the data construction, model architecture, training procedure, and hyperparameters, while the remaining appendices report additional calibration, temperature, search-budget, projection, optimization, population-transfer, game-level strength, endgame, and engine-sensitivity analyses.

REFERENCES

Max Abouchar. ChessLens Core: Open-source chess analysis and move classification, 2026. URL https://github.com/SailingSF/chesslens-core. Software repository.

Jure Andolšek, Maj Zirkelbach, Matej Černe, and Jana Krivec. Checkmate the stress: How sleep quality and engagement shape daily cognitive adaptability in elite chess players. *Applied Cognitive Psychology*, 39(6):e70146, 2025. doi: https://doi.org/10.1002/acp.70146.

Murray Campbell, A. Joseph Hoane, and Feng hsiung Hsu. Deep blue. *Artificial Intelligence*, 134 (1):57–83, 2002. ISSN 0004-3702. doi: 10.1016/S0004-3702(01)00129-1.

Chess.com. How are moves classified? What is a blunder or brilliant move?, February 2026. URL https://support.chess.com/en/articles/8572705-how-are-moves-classified-what-is-a-blunder-or-brilliant-etc. Documentation of the Classification V2 expected-points model.

Imre Csiszár. I-divergence geometry of probability distributions and minimization problems. *The Annals of Probability*, 3(1):146–158, 1975.

Arpad E Elo. *The Rating of Chessplayers, Past and Present*. Arco Pub., 1978.

Kuzman Ganchev, João Graça, Jennifer Gillenwater, and Ben Taskar. Posterior regularization for structured latent variable models. *Journal of Machine Learning Research*, 11(67):2001–2049, 2010. URL http://jmlr.org/papers/v11/ganchev10a.html.

Mark E. Glickman. Dynamic paired comparison models with stochastic variances. *Journal of Applied Statistics*, 28(6):673–689, 2001. doi: 10.1080/02664760120059219.

Jean-Bastien Grill, Florent Altché, Yunhao Tang, Thomas Hubert, Michal Valko, Ioannis Antonoglou, and Remi Munos. Monte-Carlo tree search as regularized policy optimization. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 3769–3778. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/grill20a.html.

Athul Paul Jacob, David J Wu, Gabriele Farina, Adam Lerer, Hengyuan Hu, Anton Bakhtin, Jacob Andreas, and Noam Brown. Modeling strong and human-like gameplay with KL-regularized search. In *International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 9695–9728. PMLR, Jul 2022. URL https://proceedings.mlr.press/v162/jacob22a.html.

Steffen Künn, Christian Seel, and Dainis Zegners. Cognitive performance in remote work: Evidence from professional chess. *The Economic Journal*, 132(643):1218–1232, 2022. doi: 10.1093/ej/ueab094.

Steffen Künn, Juan Palacios, and Nico Pestel. Indoor air quality and strategic decision making. *Management Science*, 69(9):5354–5377, 2023. doi: 10.1287/mnsc.2022.4643.

Lichess. Scalachess divider: Opening, middlegame, and endgame phase heuristic, 2026. URL https://github.com/lichess-org/scalachess/blob/736daea6b3fc3dddebbc69f3bd8463dc13cc198a/core/src/main/scala/Divider.scala.

Reid McIlroy-Young, Siddhartha Sen, Jon Kleinberg, and Ashton Anderson. Aligning superhuman AI with human behavior: Chess as a model system. In *Proceedings of the 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1677–1687. Association for Computing Machinery, 2020. doi: 10.1145/3394486.3403219.

Reid McIlroy-Young, Russell Wang, Siddhartha Sen, Jon M. Kleinberg, and Ashton Anderson. Learning personalized models of human behavior in chess. *CoRR*, 2020. doi: 10.48550/arXiv.2008.10086.

Reid McIlroy-Young, Yu Wang, Siddhartha Sen, Jon Kleinberg, and Ashton Anderson. Detecting individual decision-making style: Exploring behavioral stylometry in chess. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 24482–24497. Curran Associates, Inc., 2021.

Reid McIlroy-Young, Russell Wang, Siddhartha Sen, Jon Kleinberg, and Ashton Anderson. Learning models of individual behavior in chess. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1253–1263. Association for Computing Machinery, 2022. doi: 10.1145/3534678.3539367.

Daniel Monroe, George Eilender, Philip Chalmers, Zhenwei Tang, and Ashton Anderson. Chessformer: A unified architecture for chess modeling. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=2ltBRzEHyd.

Easwar Gnana Hari Sekar and Roger Jin. Human-aligned chess ai: A multitask transformer for humanlike decision-making. In *IEEE Conference on Artificial Intelligence (CAI)*, pp. 1230–1234. IEEE, 2025. doi: 10.1109/CAI64502.2025.00213.

Claude E Shannon. Xxii. programming a computer for playing chess. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 41(314):256–275, 1950. doi: 10.1080/14786445008521796.

David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, Timothy Lillicrap, Karen Simonyan, and Demis Hassabis. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419):1140–1144, 2018. doi: 10.1126/science.aar6404.

David Smerdon. The effect of masks on cognitive performance. *Proceedings of the National Academy of Sciences*, 119(49):e2206528119, 2022. doi: 10.1073/pnas.2206528119.

Anthony Strittmatter, Uwe Sunde, and Dainis Zegners. Life cycle patterns of cognitive performance over the long run. *Proceedings of the National Academy of Sciences*, 117(44):27255–27261, 2020. doi: 10.1073/pnas.2006653117.

Zhenwei Tang, Difan Jiao, Reid McIlroy-Young, Jon Kleinberg, Siddhartha Sen, and Ashton Anderson. Maia-2: A unified model for human-AI alignment in chess. In *Advances in Neural Information Processing Systems*, volume 37, pp. 20919–20944. Curran Associates, Inc., 2024. doi: 10.52202/079017-0659.

Zhenwei Tang, Difan Jiao, Eric Xue, Reid McIlroy-Young, Jon Kleinberg, Siddhartha Sen, and Ashton Anderson. Learning to imitate with less: Efficient individual behavior modeling in chess. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=iw4kjcw319.

Alan Turing. Chess (1953). In *The Essential Turing*. Oxford University Press, 09 2004. ISBN 9780198250791. doi: 10.1093/oso/9780198250791.003.0023.

Edwin Zhang, Vincent Zhu, Naomi Saphra, Anat Kleiman, Benjamin L. Edelman, Milind Tambe, Sham Kakade, and Eran Malach. Transcendence: Generative models can outperform the experts that train them. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 86985–87012. Curran Associates, Inc., 2024. doi: 10.52202/079017-2760.

Yiming Zhang, Athul Paul Jacob, Vivian Lai, Daniel Fried, and Daphne Ippolito. Human-aligned chess with a bit of search. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=bc2H72hGxB.

A Model Implementation Details

A.1 Training Data and Model Construction

Our base models follow the CHESSFORMER architecture and training setup (Monroe et al., 2026). We replicate the board tokenization, source-destination move head, value head, rating conditioning, and geometric attention bias (GAB) used by CHESSFORMER. The move head defines the policy over legal moves, while the value head predicts the game outcome from the current position; both outputs are used later in our policy-correction experiments.

Unlike the strongest CHESSFORMER setting, we do not use game history. This is a deliberate modeling choice. History is useful when the objective is to predict the next move played in a particular game, but it also introduces player- and game-specific context that should not affect a single-position judgment. Our long-term target is objective position-level difficulty and performance assessment, e.g., whether a player of rating r played well in position s. For that task, the relevant input should be the position and rating context, not the path by which the position arose.

The training data is built from public Lichess standard game dumps from January 2023 through July 2025, keeping blitz games only and sampling at most 32 positions from each selected game. We follow the CHESSFORMER chunked rating-balancing procedure, in which games are grouped into 22 rating bins and up to 10 games per bin are selected from each 20,000-game chunk (Monroe et al., 2026). One implementation detail is not fully specified in the published description: for underfilled extreme-rating bins, our reconstruction tops up nonempty bins by duplicating sampled examples from the same bin. We require clock annotations and exclude the severe time-scramble tail by stopping extraction after the post-move clock falls below 30 seconds. For evaluation, we use the ALLIE 2022 held-out test set used by ALLIE and CHESSFORMER (Zhang et al., 2025; Monroe et al., 2026); after the standard opening and clock filters, this contains 884,049 positions.

We train two no-history model scales: a 5-million-parameter model and a 23-million-parameter model, which we refer to as 5M and 23M throughout the paper. On the ALLIE test set, 5M and 23M obtain $53.7 \pm 0.1\%$ and $54.4 \pm 0.1\%$ top-1 move accuracy, respectively. These numbers are close to, but not identical with, CHESSFORMER's reported no-history results; its 5M model reports approximately 54.0% on the no-history test. We attribute the remaining ≈ 0.3 percentage-point discrepancy to training details that are not fully specified in the published paper. Direct comparison of model scales shows that the larger model achieves higher sampled playing strength alongside top-1 accuracy, motivating its use in the main experiments (Table 6). Scaling size beyond 23M parameters did not produce meaningfully better results. This is supported by results from CHESSFORMER paper where their largest 79M model produces diminishing returns on top-1 accuracy (only about 0.5% improvement in 7-move history model). Thus, we perform our experiments with the 23M model.

Both model sizes are trained with batch size 512, AdamW, mixed precision, learning rate $5 \cdot 10^{-5}$, minimum learning rate 10^{-5}, weight decay 10^{-6}, gradient clipping at 3.5, 1,000 warmup steps, and a cosine cycle of 50,000 optimizer steps. The promotion and value losses have weights 0.1. Training was run on a single NVIDIA A100 GPU and took approximately 8 hours for the 5M model and a day for the 23M model. For the results reported here, both the 5M and 23M models use the checkpoint after 1.5B training positions, which corresponds to approximately 2.93M optimizer steps at batch size 512. Table 5 summarizes the model-specific training settings.

B Additional Raw-Policy Analyses

B.1 Cross-Entropy and Calibration

CE evaluates distributions and is a proper scoring rule, but it does not directly measure calibration. Its absolute value depends on both model mismatch and the entropy of human choice. These components vary strongly with skill. In our data, the raw rating-1200 policy has CE approximately 1.48, whereas the raw rating-2400 policy has CE approximately 1.22. This does not imply that the 1200 policy is worse calibrated: the lower-rated raw policy reproduces its human move-quality target, while the higher-rated one fails to do so.

The distinction is even clearer at lower temperatures. Strengthening the rating-2400 policy with e.g. $T = 0.55$ worsens CE from approximately 1.22 to 1.42, correctly indicating reduced predictive

Table 5: Main no-history model training settings. Both models use 8 Transformer blocks, head dimension 32, feed-forward expansion factor 2, batch size 512, AdamW, mixed precision, and the shared optimizer settings described in the text.

Setting	5M	23M
d_{model}	256	512
Attention heads	8	16
d_{ff}	512	1024
GAB hidden sizes	$32, 64, 64$	$32, 128, 128$
GAB average pooling	yes	no
Training positions	1.50B	1.50B
Optimizer steps	2.93M	2.93M

Table 6: Effect of model scale on top-1 accuracy, mHPD-Err, and sampled playing strength. For each rating r, held-out metrics are computed on 20,000 ALLIE test positions with active-player rating in $[r-50, r+50]$. The match-score column reports 23M vs. 5M results from the 23M perspective over 1,000 games, with both models conditioned on the fixed rating r. Values are reported with 95% CIs.

	Top-1 acc. (%)		mHPD-Err (pp) ↓		Match score (%)
Rating r	5M	23M	5M	23M	23M vs. 5M
1200	51.7 ± 0.7	52.1 ± 0.7	1.1 ± 0.5	1.4 ± 0.5	52.8 ± 3.1
1600	54.1 ± 0.7	54.6 ± 0.7	0.7 ± 0.5	0.7 ± 0.5	54.0 ± 3.1
2000	55.6 ± 0.7	56.5 ± 0.7	0.6 ± 0.5	0.5 ± 0.5	59.3 ± 3.0
2400	57.8 ± 0.7	59.0 ± 0.7	0.4 ± 0.4	0.3 ± 0.4	60.0 ± 3.0

likelihood, but this value remains lower than the CE of the well-behaved raw rating-1200 policy. Meanwhile, mHPD-Err rises to approximately 16.0 pp. Thus, CE remains an important likelihood metric, but its absolute value does not provide a common scale for probability calibration across populations with different intrinsic uncertainty.

B.2 Move-Class Distributions Across Ratings

Table 7 reports the complete human and raw-policy move-quality distributions across rating cohorts. The high-rating deficit is driven increasingly by too little mass on best moves and too much on mistakes and blunders; at the lowest ratings, the mismatch is small and partly reversed.

Table 7: Human move and raw-policy move-quality distributions by rating bucket. H denotes human moves and M denotes the raw model policy. Each bucket is classified using its corresponding rating. Class rows are percentages.

Rating	1000		1200		1400		1600		1800		2000		2200		2400		2600	
n	46,041		45,454		47,216		52,949		53,286		54,204		52,970		48,223		13,862	
	H	M	H	M	H	M	H	M	H	M	H	M	H	M	H	M	H	M
Best	37.63	38.26	38.66	38.71	38.92	38.73	38.76	38.72	39.68	39.08	40.81	39.57	42.28	40.11	45.03	41.56	48.40	42.51
Excellent	20.64	20.80	20.42	20.55	21.34	21.32	21.74	21.64	21.65	21.45	20.57	20.39	20.09	19.92	19.10	18.73	18.92	18.25
Good	15.69	15.74	15.58	15.92	15.50	15.72	14.04	14.14	13.85	13.80	13.96	14.02	14.16	14.21	14.05	14.22	13.67	13.95
Inaccuracy	10.87	10.83	10.97	10.87	10.23	10.17	9.94	9.84	9.56	9.67	9.40	9.69	9.62	9.81	9.15	9.75	8.12	9.72
Mistake	8.29	7.88	8.10	7.87	8.08	8.02	8.70	8.61	8.69	8.72	8.85	9.05	8.45	9.01	7.95	8.97	6.88	8.97
Blunder	6.89	6.49	6.28	6.07	5.91	6.05	6.84	7.04	6.57	7.28	6.41	7.28	5.41	6.92	4.72	6.76	4.01	6.60
CPL	94.42	91.25	87.00	86.28	76.87	77.48	72.36	73.72	63.12	66.80	56.76	61.82	47.33	55.24	39.29	49.56	33.39	46.63

B.3 Rating-Conditioned Strength Structure

A central question in human behavior modeling is whether rating conditioning produces the intended relative playing strengths. Ideally, this would be evaluated through large-scale matches against real human players across many rating settings, but such evaluation is expensive and difficult to

standardize. As a complementary diagnostic, we instead examine the model's internal strength structure by running tournaments between agents conditioned on different ratings and comparing the resulting score matrix with the empirical score matrix observed between real players. While this does not replace human evaluation, it provides a useful test of whether the model reproduces realistic rating-dependent performance differences.

Figure 1 shows the empirical score matrix between real Lichess blitz rating bands. Sampled games are from the open database cohorts spanning from August to December 2025. We use this matrix as a reference for the expected score of one rating band against another. The figure also illustrates that the classical Elo expectation formula (Elo, 1978),

$$E_A = \frac{1}{1 + 10^{(R_B - R_A)/400}}, \tag{17}$$

does not perfectly describe real platform data. This is unsurprising: Lichess uses the Glicko-2 (Glickman, 2001) rating system rather than classical Elo, and real platform ratings are additionally affected by time control, player selection, rating-system dynamics, and platform-specific population effects. We therefore compare model strength against empirical rating-pair scores rather than only against the idealized Elo formula.

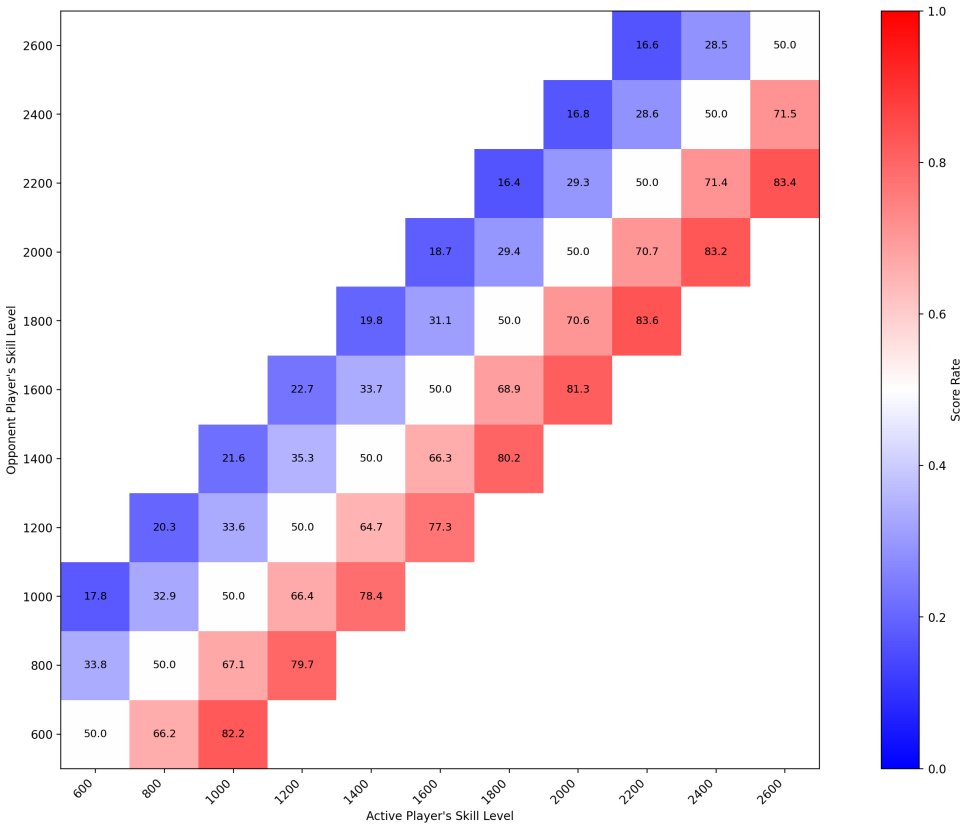

Figure 1: Empirical score matrix between real Lichess blitz rating bands. Each rating pair is estimated from approximately 2,500 games, except for the rare pairs $(2200, 2600)$ and $(2300, 2600)$, which use approximately 800 and 1,100 games, respectively.

We then run the same style of tournament internally using the raw $T = 1$ policy. Figure 2 shows the resulting model score matrix. The key pattern is that the raw policy does not separate higher rating levels strongly enough: high-rating agents conditioned on $T = 1$ do not obtain the score gaps that would be expected from the empirical Lichess reference. This provides complementary game-level evidence for the move-quality deficit identified in Section 5.1, while also showing why low aggregate HPD calibration error alone does not guarantee that sampled play realizes the intended rating structure.

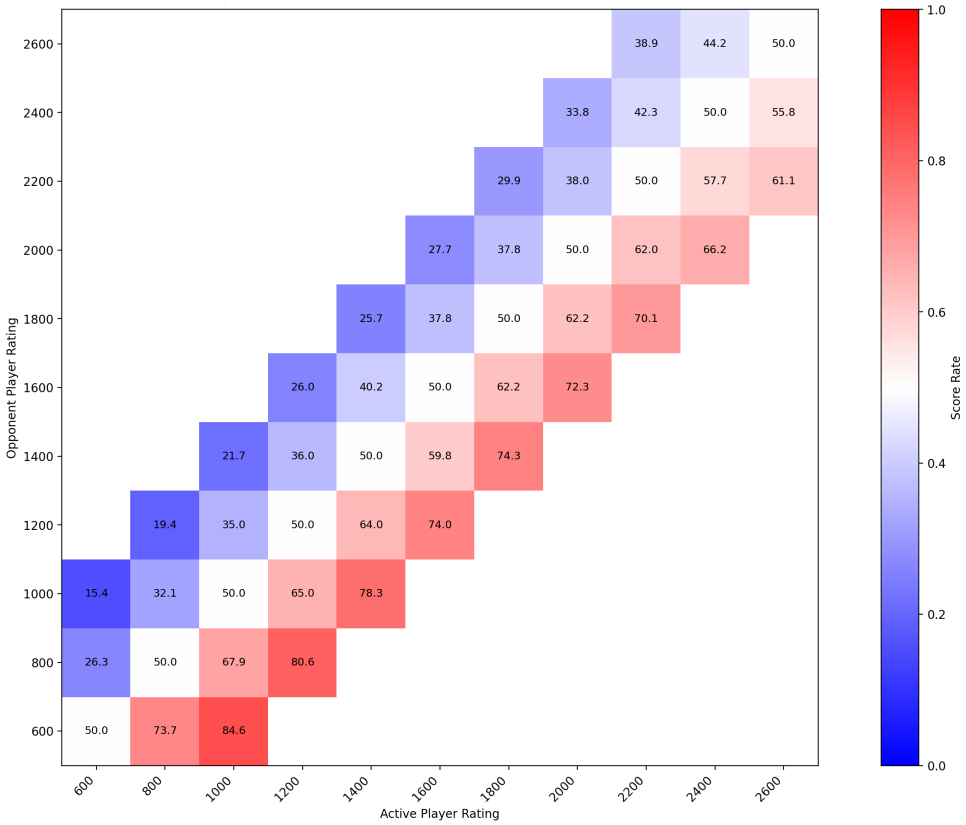

Figure 2: Internal tournament between raw $T = 1$ agents conditioned on different ratings for 23M model. Higher rating settings separate less than in the empirical Lichess score matrix, especially at stronger ratings.

B.4 Temperature Sharpening

Given a policy p_θ, temperature scaling is defined as

$$p_{\theta,T}(a \mid s, r) = \frac{p_\theta(a \mid s, r)^{1/T}}{\sum_b p_\theta(b \mid s, r)^{1/T}}. \tag{18}$$

The weak separation between raw $T = 1$ agents motivates the temperature-sharpening experiment. Instead of changing the rating conditioning or retraining the model, we modify only the inference-time sampling distribution. Lower temperatures concentrate probability mass on the model's highest-ranked moves, thereby reducing the chance of sampling lower-probability mistakes. This tests whether the model already contains stronger play in the head of its distribution, even when the full $T = 1$ distribution samples too many weak continuations.

Table 8 gives the full temperature sweep for the same-rating self-play experiment summarized in the main text. Each entry reports the score of a temperature-sharpened challenger against the raw $T = 1$ policy at the same target rating. The sweep confirms that the strength gain is not limited to deterministic top-1 play: lowering T improves score in all tested settings, with the largest gains at very low temperatures.

The same temperature transformation that strengthens sampled play also progressively changes the stochastic policy away from the human behavior it was trained to imitate. Table 9 quantifies this effect using mHPD-Err. The raw $T = 1$ policy has very low HPD coverage error, whereas increasingly aggressive sharpening produces a monotonic deterioration in calibration. Even relatively mild sharpening noticeably increases the error, and near-deterministic sampling distorts the distribution. Together with Table 8, these results illustrate the central distinction motivating our evaluation: a

Table 8: Full temperature sweep for the transcendence experiment. Each entry reports the score of the temperature-sharpened challenger against the raw $T = 1$ policy at the same target rating. Each match consisted of 500 games; 95% CIs are within ± 4.4 pp for all entries.

Target rating	$T = 0$	$T = 0.25$	$T = 0.50$	$T = 0.70$	$T = 0.80$	$T = 0.90$
600	86.9%	86.2%	80.1%	69.1%	65.3%	58.8%
1000	85.0%	84.9%	81.1%	68.5%	64.3%	60.8%
1400	83.6%	85.2%	81.3%	69.0%	66.3%	55.3%
1800	88.5%	87.6%	80.1%	70.9%	64.7%	57.6%
2200	86.8%	85.5%	79.0%	70.1%	65.9%	58.5%
2600	85.3%	86.2%	79.1%	68.4%	66.1%	57.3%

policy can become a considerably stronger chess agent while becoming a worse probabilistic model of human move selection.

Table 9: Effect of temperature sharpening on mHPD-Err. We evaluate the 23M model on 20,000 ALLIE test positions with active-player rating in $[1950, 2050]$, conditioning the model on the active-player rating of each position. Values are reported in pp with 95% CI. Lower mHPD-Err is better.

Temperature	mHPD-Err (pp) $\downarrow$
1.00	0.5 ± 0.5
0.90	3.8 ± 0.5
0.80	7.4 ± 0.5
0.70	11.3 ± 0.5
0.50	19.8 ± 0.6
0.25	31.5 ± 0.6
0.00	43.5 ± 0.7

C ADDITIONAL MHPD-ERR ANALYSES

Table 10 illustrates why evaluating multiple HPD levels is informative. For our 23M raw policy on 20,000 ALLIE test positions with active rating $1950 \leq r \leq 2050$, probability mass is highly concentrated: the 25% HPD set already has average realized mass 0.578 because the top move often exceeds the target mass. At the same time, the 90% HPD set still contains only 4.38 moves on average. Thus, different HPD levels probe different parts of a compact but nontrivial legal-move distribution.

Table 10: Average realized HPD mass and set size produced by our 23M raw policy on 20,000 ALLIE test positions with active rating $1950 \leq r \leq 2050$.

Target mass γ	Average realized mass	Average set size
0.25	0.578	1.05
0.50	0.690	1.57
0.60	0.750	1.90
0.70	0.810	2.36
0.80	0.870	3.06
0.90	0.933	4.38

We additionally test whether the reported mHPD-Err depends materially on the particular choice of evaluation levels. Replacing the main grid with denser and shifted alternatives, including $\{0.1, 0.2, \ldots, 0.9\}$, $\{0.1, 0.15, 0.2, 0.25, \ldots, 0.95\}$, and $\{0.05, 0.15, 0.25, \ldots, 0.95\}$, changes the resulting calibration scores only marginally and does not affect the qualitative conclusions. This robustness is consistent with Table 10: because the policy is discrete and concentrated, nearby low-mass thresholds frequently induce the same HPD prefix.

Finally, mHPD-Err should be interpreted specifically as an *aggregate HPD-set coverage calibration* diagnostic. It does not uniquely identify the conditional human move distribution. For example, consider two positions x_1, x_2, each with actions A, B, where a model assigns $q(A \mid x_1) = q(A \mid x_2) = 0.8$, while the true probabilities are $p^\star(A \mid x_1) = 1$ and $p^\star(A \mid x_2) = 0.6$. For HPD levels up to 0.8, average human coverage is 0.8, exactly matching the model's attained mass; above 0.8 both actions are included and both quantities equal one. Aggregate HPD error is therefore zero despite incorrect conditional probabilities. We consequently use mHPD-Err as a complementary calibration diagnostic rather than a complete measure of distributional agreement.

D Additional Policy-Correction Results

D.1 MCTS Budget Ablation

We tune c_{puct} separately at each MCTS budget on the same 5,000-position pilot set at rating 2400, whose human mean CPL is 38.57. Holding expected CPL approximately fixed isolates the effect of additional search. Table 11 shows that larger budgets consistently improve CE, mHPD-Err, Class L1, and severity alignment, although the gains diminish as the budget grows. The main paper uses $B = 50$ because it matches the principal search budget used in prior human-aligned MCTS work (Jacob et al., 2022; Zhang et al., 2025) while remaining computationally practical.

Table 11: Effect of MCTS budget at rating 2400 on the 5,000-position pilot set after independently tuning c_{puct} to match the human mean CPL (1.00, 1.44, 2.21, 3.00, 3.60, 5.60 for increasing budgets respectively).

Budget	Top-1 (%)	CE	mHPD-Err (pp)	Class L1 (pp)	Severity MAE (cp)
20	58.10	1.251	5.04	3.42	1.108
30	57.86	1.245	4.65	3.06	1.069
50	58.04	1.237	4.12	2.68	0.864
75	58.16	1.232	3.78	2.46	0.786
100	58.36	1.230	3.61	2.35	0.715
200	58.46	1.227	3.30	2.04	0.659

D.2 Move-Class Distributions by Rating

Tables 12–15 extend the rating-2500 comparison to ratings 2100, 2200, 2300, and 2400. They compare the empirical human distribution with the probability assigned by the raw model, CPL-matched MCTS, and the out-of-fold 80%-shell engine projection.

Table 12: Move-class distribution and CPL at rating 2100 in the Allie set.

	Human	Raw model	MCTS	Projection
Best (%)	41.46	39.90	41.92	41.23
Excellent (%)	20.23	20.00	20.25	20.21
Good (%)	14.22	14.21	14.07	14.27
Inaccuracy (%)	9.45	9.79	9.45	9.51
Mistake (%)	8.71	9.06	8.46	8.77
Blunder (%)	5.92	7.04	5.85	6.00
CPL	50.52	56.57	50.26	50.45

D.3 Population projection ablations

Table 16 isolates the ingredients of the final projection at rating 2400. All learned variants use five-fold out-of-fold predictions on the same 48,223 positions. The class-only variant fits the five nonredundant class-mass offsets in Equation 14; the severity-only variant fits the five class-specific CPL-contribution slopes; and the combined variant fits both. The CPL-only baseline instead uses the single feature $f_i(a) = \ell_i(a)/100$, producing a one-parameter Stockfish-CPL logit reweighting whose sole target is human mean CPL. It otherwise uses the same human prior, annotations, folds,

optimizer, and 80% shell as the final projection. To test whether the learned human prior is necessary in addition to the engine-derived features and aggregate human targets, we replace it with a uniform distribution over legal moves.

Table 13: Move-class distribution and CPL at rating 2200 in the ALLIE set.

	Human	Raw model	MCTS	Projection
Best (%)	42.28	40.11	42.97	42.09
Excellent (%)	20.09	19.92	20.26	20.06
Good (%)	14.16	14.21	14.01	14.20
Inaccuracy (%)	9.62	9.81	9.30	9.67
Mistake (%)	8.45	9.01	8.10	8.49
Blunder (%)	5.41	6.92	5.35	5.49
CPL	47.33	55.24	47.00	47.30

Table 14: Move-class distribution and CPL at rating 2300 in the ALLIE set.

	Human	Raw model	MCTS	Projection
Best (%)	43.73	40.85	44.34	43.55
Excellent (%)	19.88	19.48	19.91	19.84
Good (%)	13.90	14.01	13.72	13.95
Inaccuracy (%)	9.24	9.76	9.12	9.28
Mistake (%)	8.27	9.05	7.92	8.31
Blunder (%)	4.98	6.85	4.99	5.08
CPL	42.93	52.04	42.70	42.90

Table 15: Move-class distribution and CPL at rating 2400 in the ALLIE set.

	Human	Raw model	MCTS	Projection
Best (%)	45.03	41.56	45.62	44.81
Excellent (%)	19.10	18.73	19.27	19.08
Good (%)	14.05	14.22	13.88	14.09
Inaccuracy (%)	9.15	9.75	8.97	9.19
Mistake (%)	7.95	8.97	7.63	7.98
Blunder (%)	4.72	6.76	4.63	4.85
CPL	39.29	49.56	39.08	39.26

The CPL-only projection matches human mean CPL essentially exactly, but its Class L1 and Severity MAE remain larger than under the full shell-preserving projection. Thus, the improvement is not explained by access to Stockfish CPL or by exponential reweighting alone: class-frequency and class-conditioned-severity targets control aspects of the error distribution that its scalar mean leaves unidentified. The full projection slightly increases mHPD-Err relative to CPL-only while improving CE and sharply improving both move-class and severity alignment.

The two feature families correct different failure modes. Class offsets nearly recover the aggregate class distribution but leave a $+2.49$ CPL error and large within-class severity error. Severity slopes recover mean CPL but leave a 2.40-pp class mismatch. Their combination aligns both, while the 80% shell reduces mHPD-Err from 1.73 to 0.98 pp with only a small CE cost. The uniform-prior variant also matches CPL and severity on average, but its top-1 accuracy falls by 16.76 pp and CE rises to 2.052. Engine annotations therefore supply the population-level quality correction, whereas the learned prior remains essential for selecting human-plausible moves within those quality regions.

D.4 HPD PRESERVATION ABLATIONS

The unconstrained class+severity projection matches the human move-quality moments but can transfer probability between the high-density region of the raw policy and its tail, increasing mHPD-Err. We therefore jointly fit the projection while preserving raw HPD-shell masses. Table 17 compares this approach with alternative shell definitions and with a simpler post-hoc correction that restores shell masses after fitting.

Table 16: Component ablation at rating 2400. Human denotes the learned raw-policy prior and Uniform denotes a construction with a uniform prior over legal moves. ΔCPL is relative to human CPL.

Prior	Fitted features	Top-1 (%)	CE	ΔCPL	mHPD-Err (pp)	Class L1 (pp)	Sev. MAE (cp)
Human	none (raw policy)	58.86	1.203	+10.27	0.19	7.67	2.732
Human	class	59.18	1.194	+2.49	1.71	0.48	2.493
Human	severity	59.24	1.194	-0.07	1.43	2.40	1.050
Human	class + severity	59.23	1.193	-0.06	1.73	0.49	0.066
Uniform	class + severity	42.47	2.052	-0.01	3.30	4.04	0.026
Human	CPL + shell	59.25	1.196	0.00	0.77	2.37	1.164
Human	class + severity + shell	59.22	1.195	-0.03	0.98	0.49	0.031

Table 17: HPD-preservation variants at rating 2400 on all 48,223 positions. Post-hoc rows rescale the frozen unconstrained projection without refitting; the remaining rows fit the engine correction jointly within the stated shells.

Policy	ΔCPL	CE	mHPD-Err (pp)	Class L1 (pp)	Sev. MAE (cp)
Unconstrained	-0.061	1.193	1.73	0.49	0.066
Post-hoc 50%	+4.942	1.197	0.39	4.51	1.390
Post-hoc 80%	+2.754	1.195	0.59	2.57	0.752
Post-hoc 90%	+1.771	1.194	0.95	1.73	0.509
Post-hoc 95%	+1.144	1.193	1.24	1.23	0.353
Post-hoc 99%	+0.355	1.193	1.59	0.70	0.171
Post-hoc 99.9%	+0.025	1.193	1.71	0.52	0.096
80% shell	-0.028	1.195	0.98	0.49	0.031
90% shell	-0.040	1.194	1.21	0.49	0.047
80%+90% shells	-0.031	1.197	1.03	0.49	0.036
70%+80%+90% shells	-0.023	1.201	1.04	0.48	0.028

Post-hoc preservation illustrates the trade-off directly: tighter boundaries can reduce mHPD-Err, but largely undo the CPL and class/severity correction. At the opposite extreme, a 99.9% boundary retains the CPL match but provides little calibration improvement. Joint fitting avoids this trade-off by allowing the engine correction to adapt within the preserved regions.

Among the jointly fitted variants, a single 80% shell provides the best observed calibration–likelihood balance. It reduces mHPD-Err from 1.73 to 0.98 pp relative to the unconstrained projection while changing CE only from 1.193 to 1.195 and leaving Class L1 unchanged. Moving the boundary to 90% weakens the calibration improvement, while enforcing several exact boundaries provides no corresponding benefit and can slightly worsen CE. The improvement is not confined to the preserved 80% boundary: Appendix D.5 reports HPD error separately at every coverage level used by mHPD-Err.

D.5 HPD Error by Coverage Level

The aggregate mHPD-Err reduction produced by the 80% shell is not driven by the preserved boundary alone. Table 18 reports calibration error separately at every evaluation level. The shell-constrained projection improves HPD coverage throughout the distribution, with particularly clear gains in the middle and upper portions of the policy support. Preserving a broad head–tail allocation therefore also stabilizes lower-level prefixes after the within-shell engine reweighting.

D.6 Projection Optimization Diagnostics

Constraint feasibility is assessed on the effective support after fixing any prior mass without a usable engine annotation. A fold is accepted only if the dual solver converges and the largest absolute difference between its achieved and target feature moments is below 10^{-6} in the scaled coordinates of Equation 15; otherwise the fit is rejected rather than silently relaxing the targets. All 50 folds

Table 18: HPD calibration error in pp at each evaluation level for the unconstrained and 80%-shell class+severity projections at rating 2400.

HPD level	0.25	0.50	0.60	0.70	0.80	0.90
Class+severity	2.12	2.12	2.04	1.86	1.37	0.84
Class+severity, 80% shell	1.64	1.52	1.32	0.80	0.37	0.26

across the five ALLIE and five broader-Lichess cohorts converged in at most 10 Newton iterations, and the largest observed residual was 1.7×10^{-8}.

D.7 ADAPTIVE ENGINE-MOVE SELECTION

The main position-level experiments evaluate every legal move with the engine before applying the policy projection. This is unnecessary when the human-model distribution is concentrated: many legal moves have probabilities near zero and can have only a negligible effect on the projected policy, despite incurring the same engine-evaluation cost as plausible moves. We therefore investigate whether engine computation can be reduced by restricting evaluation to moves that cover nearly all of the raw policy mass.

For a coverage threshold ρ, legal moves are sorted by raw model probability and we evaluate the smallest prefix whose cumulative mass reaches ρ. We also include an unrestricted engine-best move if it is absent from this prefix, ensuring that a strong move overlooked by the model remains available. Moves outside the evaluated set retain their original probability, while the frozen fold-specific 80%-shell projection is applied to the evaluated support. The projection parameters are neither refitted nor retuned for any value of ρ; the ablation isolates only the effect of reducing engine coverage.

Table 19 reports the fine sweep at rating 2400. Increasing coverage from 99.0% to 99.9% raises the average number of engine-evaluated moves from 8.46 to 14.41, but reduces the residual CPL error from $+0.44$ to $+0.07$ and progressively improves class and severity alignment. Across the sweep, CE and mHPD-Err remain essentially unchanged. At 99.9% coverage, the method evaluates 56% fewer moves than exhaustive evaluation (14.41 versus 32.77 on average), with only small residual differences in class L1 and severity MAE.

Table 19: Adaptive engine-move selection at rating 2400 on all 48,223 positions, using frozen five-fold parameters. Each policy evaluates the smallest raw-policy prefix reaching coverage ρ, augmented with an unrestricted engine-best move.

ρ	Avg. moves	ΔCPL	CE	mHPD-Err (pp)	Class L1 (pp)	Sev. MAE (cp)
99.0%	8.46	+0.44	1.195	0.98	0.57	0.244
99.1%	8.69	+0.41	1.195	0.97	0.56	0.231
99.2%	8.96	+0.37	1.195	0.98	0.55	0.216
99.3%	9.27	+0.34	1.195	0.98	0.55	0.200
99.4%	9.64	+0.31	1.195	0.98	0.54	0.185
99.5%	10.08	+0.27	1.195	0.98	0.53	0.171
99.6%	10.63	+0.23	1.195	0.98	0.53	0.155
99.7%	11.37	+0.18	1.195	0.98	0.52	0.136
99.8%	12.45	+0.13	1.195	0.98	0.51	0.111
99.9%	14.41	+0.07	1.195	0.98	0.50	0.079
All	32.77	-0.03	1.195	0.98	0.49	0.031

Thus, 99.9% adaptive coverage provides a practical computation–fidelity trade-off: it allocates engine work according to how concentrated the model distribution is in each position while preserving essentially all of the exhaustive policy's measured fidelity. We use this 99.9% setting for the game-level tournaments in Section 5.4.

E LARGE-SCALE POPULATION AND GAME-LEVEL ANALYSES

The ALLIE cohorts are sufficient for controlled comparison with prior work, but are comparatively small and contain predominantly similarly rated opponents. To test population transfer at larger scale, we construct balanced rating-pair cohorts from Lichess blitz games played between August and December of 2025. We apply the same position filters as in the main experiments: the first 10 plies are excluded, and positions after either player falls below 30 seconds are not used. The resulting Stockfish-annotated cohorts contain 521,908, 492,213, 437,350, 410,096, and 321,836 positions at ratings 2100 through 2500, respectively.

E.1 OPPONENT STRENGTH CHANGES POSITION DIFFICULTY

Table 20 stratifies the rating-2400 cohort by the signed opponent-rating gap. Human CPL is lowest against weaker opponents and rises from 34.60 against opponents at least 300 points lower to 36.84 against similarly rated opponents. It does not increase further against opponents 100–199 points stronger. This may reflect selection or motivational effects—weaker players may approach games against stronger opponents differently—but the observational data cannot identify the mechanism. We do not use the 200-or-more-stronger group to infer this trend for rating 2400 because it contains only 13,807 positions, far fewer than the other buckets.

Table 20: Human CPL for rating-2400 players in the broader Lichess cohort, stratified by opponent rating. The gap is opponent rating minus player rating.

Opponent gap	N	Human CPL
≤ -300	100,478	34.60
$[-299, -200]$	60,047	35.43
$[-199, -100]$	80,137	35.57
$[-99, 99]$	73,438	36.84
$[100, 199]$	82,189	36.84

Human CPL in the broader Lichess cohort is lower than in the corresponding ALLIE cohort. This may reflect sampling differences between the datasets or temporal changes in the Lichess rating pool, such as rating deflation.

E.2 TRANSFER OF THE ALLIE-FITTED PROJECTION

Table 21 applies each rating's final ALLIE-fitted 80%-shell parameter vector to the broader Lichess cohort without refitting. The correction transfers in the useful direction at every rating: it reduces the raw CPL error by 5.66–13.47 points and improves CE and Class L1. Residual CPL errors nevertheless range from $+4.05$ at rating 2100 to $+1.88$ at rating 2500. Thus, a population projection faithfully reproduces the aggregate targets on which it is calibrated, but those targets need not transfer unchanged to a population with a different opponent mixture and sampling period.

E.3 BROADER LICHESS POPULATION PROJECTION

We refit the class+severity 80%-shell projection separately within each broader rating cohort and evaluate it by five-fold cross-validation. Table 21 shows that recalibration on the target population removes the residual move-quality mismatch left by transferring the ALLIE-fitted correction. It recovers the human class distribution and within-class error severity while also improving predictive likelihood over the raw policy. The accompanying increase in mHPD-Err reflects the remaining tension between changing move quality and preserving the raw policy's aggregate coverage calibration, even when its broad 80% head–tail structure is held fixed. Together, the transfer and within-cohort results show that the projection method generalizes, but its target moments must represent the population in which it will be used. Tables 22–26 show the corresponding move-class distributions and CPL values. Together, these results show that the projection transfers across populations but is most accurate when calibrated on representative data. Because a rating-level vector still averages over opponent-dependent position difficulty, conditioning the targets or tilt on opponent rating is a natural extension.

Table 21: Transfer and within-cohort projection on the broader Lichess (BL) rating-pair data. The ALLIE projection uses the frozen rating-specific vector fitted on the full ALLIE cohort. Each within-cohort 80%-shell projection is fitted on four broader-Lichess folds and applied to the held-out fold. N is the pooled number of positions.

Rating	N	Policy	Top-1 (%)	CE	ΔCPL	mHPD-Err (pp)	Class L1 (pp)	Sev. MAE (cp)
2100	521,908	Raw	57.21	1.272	+9.71	0.23	6.96	3.126
		ALLIE projection	57.45	1.266	+4.05	0.16	4.00	1.150
		BL projection	57.50	1.266	+0.00	0.47	0.50	0.006
2200	492,213	Raw	57.62	1.258	+11.12	0.05	8.48	3.356
		ALLIE projection	58.01	1.249	+3.68	0.35	4.42	0.930
		BL projection	58.08	1.249	-0.01	0.92	0.48	0.009
2300	437,350	Raw	58.01	1.239	+12.38	0.29	10.11	3.295
		ALLIE projection	58.57	1.226	+3.44	0.71	4.21	1.152
		BL projection	58.61	1.227	+0.00	1.31	0.46	0.012
2400	410,096	Raw	58.47	1.220	+13.46	0.47	11.53	3.130
		ALLIE projection	59.17	1.203	+3.04	0.93	4.58	1.547
		BL projection	59.21	1.204	+0.01	1.67	0.42	0.014
2500	321,836	Raw	58.80	1.209	+15.35	0.74	14.36	2.982
		ALLIE projection	59.89	1.184	+1.88	1.48	4.51	1.231
		BL projection	59.92	1.185	+0.01	2.22	0.47	0.010

Table 22: Move-class distribution and CPL at rating 2100 in the broader Lichess set.

	Human	Raw model	Projection
Best (%)	42.65	39.65	42.43
Excellent (%)	21.25	20.77	21.22
Good (%)	13.96	14.20	14.02
Inaccuracy (%)	9.35	9.87	9.40
Mistake (%)	7.91	8.90	7.95
Blunder (%)	4.88	6.60	4.98
CPL	46.12	55.83	46.12

Table 23: Move-class distribution and CPL at rating 2200 in the broader Lichess set.

	Human	Raw model	Projection
Best (%)	44.00	40.27	43.76
Excellent (%)	20.89	20.37	20.89
Good (%)	14.04	14.33	14.09
Inaccuracy (%)	9.05	9.79	9.11
Mistake (%)	7.65	8.88	7.69
Blunder (%)	4.37	6.35	4.47
CPL	42.14	53.26	42.13

Table 24: Move-class distribution and CPL at rating 2300 in the broader Lichess set.

	Human	Raw model	Projection
Best (%)	45.20	40.83	44.96
Excellent (%)	20.74	20.04	20.75
Good (%)	13.75	14.22	13.79
Inaccuracy (%)	8.81	9.66	8.86
Mistake (%)	7.33	8.80	7.37
Blunder (%)	4.17	6.44	4.27
CPL	38.39	50.77	38.39

Table 25: Move-class distribution and CPL at rating 2400 in the broader Lichess set.

	Human	Raw model	Projection
Best (%)	46.42	41.38	46.21
Excellent (%)	20.38	19.66	20.39
Good (%)	13.72	14.19	13.77
Inaccuracy (%)	8.58	9.60	8.61
Mistake (%)	7.11	8.78	7.13
Blunder (%)	3.78	6.39	3.89
CPL	35.80	49.26	35.81

Table 26: Move-class distribution and CPL at rating 2500 in the broader Lichess set.

	Human	Raw model	Projection
Best (%)	48.39	42.02	48.15
Excellent (%)	20.27	19.46	20.30
Good (%)	13.28	14.05	13.32
Inaccuracy (%)	8.19	9.43	8.23
Mistake (%)	6.51	8.66	6.53
Blunder (%)	3.35	6.38	3.47
CPL	31.81	47.16	31.81

E.4 GAME-LEVEL RATING STRUCTURE

We compare game-level strength across ratings 2100–2500 by playing 1,000 games for each rating pair, with colors balanced evenly, and compare the score of the lower-rated agent with the corresponding empirical Lichess score rate. Table 27 reports tournaments for both the ALLIE-fitted projection and the projection fitted to the larger Lichess cohorts.

Table 27: Game-level score rates for rating pairs. Entries are the percentage of available points scored by the lower-rated player or agent. Each agent tournament contains 1,000 games with equal colors and has an approximate 95% confidence interval of ± 2.8 percentage points. The empirical Lichess intervals are approximately ± 1.5–2.0 percentage points.

Rating pair	Empirical Lichess	ALLIE projection	Broader-Lichess projection
2100–2200	39.90	41.70	42.35
2100–2300	28.90	35.20	34.25
2100–2400	21.30	28.50	27.60
2100–2500	15.50	21.10	23.70
2200–2300	38.80	43.20	43.90
2200–2400	28.60	37.15	34.30
2200–2500	21.50	28.05	30.00
2300–2400	39.60	42.80	46.30
2300–2500	29.10	35.80	35.00
2400–2500	37.10	39.40	40.60

The broader-Lichess projection produces rating-pair scores similar to those of the ALLIE projection. Consequently, the tournament discrepancy cannot be attributed solely to the smaller ALLIE calibration sample being insufficiently representative. This result shifts attention from the marginal calibration cohorts to behavior that is absent from both projection settings.

Both the model training data and the projection calibration positions apply a strict requirement that both players have at least 30 seconds remaining, thereby excluding time-scramble blunders. In blitz chess, this filter disproportionately removes endgames, which are frequently played with little time. The learned policy and projection targets may therefore omit practical endgame behavior that materially affects game outcomes. This motivates a phase-specific analysis of the tournament games.

E.5 GAME-PHASE ERROR PROFILE

We divide each tournament game into opening, middlegame, and endgame using the Lichess phase heuristic (Lichess, 2026). The middlegame begins at the first position in which at most ten non-pawn, non-king pieces remain, either player's back rank contains fewer than four pieces, or Lichess's local piece-mixedness score exceeds 150. Once a middlegame has been identified, the endgame begins at the first position with at most six non-pawn, non-king pieces. Table 28 reports post-hoc Stockfish CPL for the 1,000-game rating-2200 versus rating-2400 tournament. Each played move is evaluated at depth 15 by an unrestricted Stockfish-18 root search and a second search restricted to that move, using two engine threads. These settings produce slightly different absolute CPL values from the main annotation pipeline, but the conclusion of substantially elevated endgame CPL is unchanged.

Table 28: Played-move CPL by Lichess game phase in the 1,000-game broader-Lichess projection tournament between rating-2200 and rating-2400 agents. N counts moves with defined CPL; moves involving a mate score are excluded.

	Rating 2200			Rating 2400		
Phase	N	Mean CPL	Median CPL	N	Mean CPL	Median CPL
Opening	11,218	20.05	6	11,231	17.33	4
Middlegame	15,823	47.71	14	15,836	42.19	10
Endgame	17,349	131.34	11	17,430	123.75	6
All phases	44,390	73.40	10	44,497	67.86	7

Mean CPL rises sharply in the endgame for both agents, while the much smaller medians show that this increase is driven by a heavy tail of errors rather than uniformly poor endgame play. The phase analysis does not establish that the 30-second filter causes the tournament-score discrepancy, but it identifies a plausible mechanism: many blitz games are decided in practical endgames, where stronger players can continue posing problems and exploiting errors that are poorly represented in the filtered training and calibration positions.

To explore this move-quality deficit further, we apply the same phase definition to the broader Lichess annotation cohorts after the 30-second cutoff. Table 29 compares human played-move CPL with the expected CPL of the fitted 80%-shell projection. The projection follows human mean CPL closely in every phase. This agreement is expected rather than evidence that the agent has learned complete endgame play, because both the underlying policy and the projection targets are estimated from positions subject to the same 30-second filter.

Table 29: Move share and mean CPL by Lichess game phase in the 30-second-filtered broader-Lichess cohorts. Human denotes played-move CPL and Projection denotes expected CPL under the fitted 80%-shell engine projection.

	Rating 2200			Rating 2400		
Phase	Moves (%)	Human CPL	Projection CPL	Moves (%)	Human CPL	Projection CPL
Opening	22.6	24.58	24.64	22.4	19.81	19.91
Middlegame	51.5	42.83	43.12	52.1	37.08	37.39
Endgame	25.9	56.96	56.16	25.5	47.83	47.08

The phase composition reveals the more important effect of the cutoff. To measure the filtering effect directly in human data, we replay the source games and assign phases before applying the clock cutoff. Table 30 counts every rating-conditioned position after the opening exclusion, including positions subsequently removed by the clock rule. The cutoff removes 20–25% of all positions and 45–53% of endgame positions. Consequently, the endgame share falls from 39–41% before filtering to 26% after filtering. Endgames are therefore underrepresented both in model training and in projection calibration, making degraded endgame behavior unsurprising. Reproducing exact blitz rating-pair scores may require training on positions below the cutoff; however, doing so would also model time trouble and time-scramble chess, which is outside our objective of modeling deliberate

move choice. How to preserve this objective while matching the practical endgame skill of blitz populations remains an open question.

Table 30: Human positions removed by the 30-second cutoff in the broader Lichess cohorts. Full counts include all rating-conditioned positions after the 10-ply opening exclusion. Endgame removal is reported relative to the full endgame count. Rm denotes removed.

Rating	Full positions	Rm	Rm (%)	Full endgames	Rm endgames	Rm endgames (%)
2100	656,598	134,690	20.51	254,455	115,081	45.23
2200	629,358	137,145	21.79	244,696	117,327	47.95
2300	571,089	133,739	23.42	226,093	114,741	50.75
2400	539,275	129,179	23.95	215,759	111,281	51.58
2500	429,346	107,510	25.04	175,285	93,108	53.12

E.6 Deployment Against Human Opponents

As a small external check, we deployed a rating-2000-conditioned agent that samples from the raw policy with temperature $T = 0.55$[2]. This configuration was selected near the beginning of the study as a plausible approximation to rating-2000 play, independently of the phase analysis. To avoid estimates being influenced by the sparsely represented rating tails, we retain opponents rated from 1600 through 2200. The resulting 135 completed blitz games have an observed opponent-rating range of 1623–2125 and mean 1935; the agent scored 54.81% (68 wins, 12 draws, and 55 losses). This record is broadly consistent with the intended strength, but it is not a precise rating estimate: the sample is small, contains repeated opponents and both rated and casual games.

Table 31 applies the same post-hoc phase analysis to these games without the 30-second cutoff. The agent has lower mean CPL than its opponents in the opening and middlegame, but higher mean CPL in the endgame (122.07 versus 110.05), despite not being subject to human time pressure. This result does not establish a causal effect of the training cutoff, nor does it imply that the pooled opponents constitute a rating-2000 reference population. It nevertheless provides supporting evidence that approximately matching aggregate playing strength can conceal a different phase profile, with the clearest deficit appearing in practical endgames.

Table 31: Played-move CPL by Lichess game phase for the rating-2000-conditioned agent ($T = 0.55$) and human opponents rated 1600–2200 over 135 blitz games. N counts moves with defined CPL; moves involving a mate score are excluded. Games are unfiltered and contain time-pressure positions.

	Agent		Human opponents	
Phase	N	Mean CPL	N	Mean CPL
Opening	1,429	14.89	1,430	17.03
Middlegame	1,909	47.24	1,903	51.32
Endgame	2,122	122.07	2,104	110.05
All phases	5,460	67.85	5,437	65.03

F Engine parameters and compute

All move-quality annotations use Stockfish 18 at full strength through the UCI interface. We disable `UCI_LimitStrength`, use one search thread and 128 MB of hash, and stop each search at depth 15. No additional node or wall-clock limit is imposed. Engine version is important because both centipawn evaluations and the engine's Win-Draw-Loss (WDL) model can change between Stockfish releases. Stockfish does not itself output CPL: for non-mate scores we compute it as the difference between the best root score and the candidate root score, always from the perspective of the player to move. CPL is left undefined when this subtraction would cross a mate score.

[2]Lichess Bot, link hidden for anonymity

For each position, all legal root moves are evaluated together in one MultiPV search, with `MultiPV` set to the number of legal moves. The sole exception is the post-hoc tournament analysis in Table 28: it uses the same engine version and depth but two threads and paired single-PV searches, one unrestricted search for the best move and one search restricted to the played move. Annotating the 2.18 million positions across the 2100–2500 rating cohorts in the broader Lichess dataset required approximately 4300 CPU core-hours. Each worker used a single Stockfish thread and processed approximately 0.14 positions per second on average.

Table 32 gives the complete classification configuration. Native Stockfish WDL values are requested with `UCI_ShowWDL`. WDL affects only the ChessLens draw-sensitivity adjustment. Specifically, expected-points losses of at least 0.05 are multiplied by a factor that increases linearly from 1.0 when the engine draw probability is at most 0.5 to 1.3 at draw probability one. Mate evaluations map directly to expected points zero or one. The rating-conditioned expected-points mapping and the six class boundaries are defined in Section 4.2.

Table 32: Stockfish and move-classification parameters used for dataset annotations. The post-hoc tournament analysis in Table 28 uses the exception described in the text.

Setting	Value
Engine	Stockfish 18, full strength
Root search	all legal moves in one MultiPV search
Search limit	depth 15
Threads / hash	1 / 128 MB
Saved PV length	6 plies
Mate CP equivalent	100,000 (storage and ordering only)
EP sigmoid coefficient	0.00368
Rating scale	$0.85 + 0.70\,\sigma(0.01(r - 1500))$
EP-loss boundaries	0, 0.02, 0.05, 0.10, 0.20
Draw boost	threshold 0.5; maximum factor 1.3
Minimum loss for draw boost	0.05 EP

F.1 Stockfish depth sensitivity

Table 33 evaluates the complete rating-2500 Allie cohort at depths 15–18. Each entry is obtained from an independent all-legal-move MultiPV search with the hash cleared immediately beforehand; consequently, all depths are compared under the same search-state conditions. This differs from the main annotation pipeline, which reuses one engine process and its transposition table within each shard. The absolute depth-15 values therefore differ slightly from the main results and should not be interpreted as a reproduction of them; the purpose of this experiment is the controlled within-table comparison across depths.

Table 33: Sensitivity of move-quality annotations to Stockfish search depth on all 30,465 rating-2500 Allie positions. Class entries are percentages. Model denotes expected quantities under the uncorrected model policy. Every position and depth uses an independent clean-hash search.

Depth	Source	Best	Excellent	Good	Inaccuracy	Mistake	Blunder	CPL
15	Human	46.37	19.11	14.37	8.89	7.21	4.04	32.74
	Model	41.72	18.55	14.49	9.81	8.89	6.54	44.68
16	Human	46.26	19.56	13.87	9.05	7.05	4.21	33.81
	Model	41.77	18.72	14.23	9.79	8.80	6.69	46.18
17	Human	46.33	19.71	13.57	8.93	7.12	4.34	34.20
	Model	41.86	18.82	14.15	9.58	8.83	6.75	46.77
18	Human	46.51	19.38	13.88	8.72	7.18	4.33	35.20
	Model	41.87	18.85	14.03	9.60	8.82	6.83	48.08

Increasing depth from 15 to 18 raises human played-move CPL by 2.46 and raw-policy expected CPL by 3.40, showing that deeper search exposes losses that are missed by shallower evaluation. In contrast, the class distributions are stable: for both humans and the raw policy, every class changes

by less than 0.5 percentage points between depths 15 and 18. The main conclusions about the model's move-class mismatch are therefore insensitive to this depth range, although the absolute CPL scale depends modestly on engine depth.