From learned chess intuition to reliable play

Early draft v0.1 · 6 November 2026

Abstract. We describe a hybrid chess player that combines a learned move policy and outcome estimates with selective lookahead, bounded Stockfish assistance and clock-aware execution. The central research question is whether richer supervision and targeted computation improve the complete player. This early draft separates deployed mechanisms from an experimental training recipe; full-system performance results are pending.

1. The challenge

Predicting a plausible move is only part of playing chess well. A move may look natural but lose tactically; a winning position may still be difficult to finish; thinking longer may lose on time. LittlePly therefore combines a learned model, selective lookahead, targeted Stockfish assistance and strict time management. We evaluate the resulting player as a whole.

2. The mental model

1. The model proposes. Treat the game as a sequence of moves, like words in a sentence. Given the complete history, the model ranks legal next moves. This ranking is its policy. It also estimates win/draw/loss probabilities, giving search a way to compare resulting positions. These estimates are learned judgments, not guarantees.

2. Search tests alternatives when the policy is indecisive. If several moves look similarly plausible, the system explores a small tree of model-proposed moves and opponent replies. It prefers moves that lead to better estimated outcomes against the opponent's best explored response. This can overturn the first suggestion, but may miss a defense outside the shortlist.

3. Specialized checks can change the choice. Stockfish searches for mating opportunities and threats, often while the opponent is thinking. A supported mate threat can veto the chosen move; a winning continuation can replace it. Rule checks can find an immediate automatic draw when the chosen move faces a supported mating threat. In small, strongly winning endgames, a separate Stockfish conversion step may choose a more precise continuation. This conversion step is applied last.

4. The clock bounds everything. Search keeps its last fully completed answer. If time runs out, unfinished work is discarded rather than delaying the move. Assistance is bounded and may return no usable evidence; that alone does not overturn the model's choice.

Training aims to strengthen the first stage: next-move prediction learns proposals, outcome prediction supplies evaluation, future-move prediction adds a longer horizon, and board reconstruction asks the model to track the position. Whether these training ideas improve the complete player remains an experimental question. [1-3]

3. Training the model

Data. The certified union contains 26,703,686 unique games and 2,263,679,463 eligible next-move targets: original Elite (23,691,842 games), engine (172,656), and January/May/September/October 2026 additions (728,232/698,300/688,076/724,580). Human selection requires both ratings ≥2300. Preparation validates legal standard-start histories, deduplicates histories/roots, and excludes frozen evaluation histories and first-30-ply families. Engine book moves remain context, but 2,700,376 book targets are excluded. An older overlapping pool adds no games; partial March is excluded. Two passes target 4,527,358,926 retained ordinary uses, without winner-biased sampling or 80/20 resampling. [1]

Learning tasks. One token represents one complete UCI move, including promotion. Ordinary supervision predicts the next move. Auxiliary queries predict fixed-White WDL, a recorded move at offsets 2-9, or 64 occupancy cells (a8-h1) plus END, not full FEN. WDL/future task share ramps 5 to 30%, equally split; board targets 5%, subject to eligibility/rounding. Ordinary loss weights are 4/3 for the eventual winner's moves, 2/3 for the loser's and 1 for draws, normalized over the global ordinary subset per update. Board loss averages 65 labels; other auxiliary weights are unchanged. Full-vocabulary cross-entropy and independent exposure caps apply. Per-layer masks prevent auxiliary answers leaking into clean policy states. [1,2]

Architecture. The experimental model has 86,977,536 parameters: GPT-2, 12 layers, width 768, 12 heads, MLP 3072, 512 learned positions, GELU-new, LayerNorm epsilon 1e-5, no dropout and tied input/output embeddings. The official GPT-2 body and first 512 positional rows initialize it; all 1,990 token rows are fresh (seed 20261104). Vocabulary: 1,968 moves, three outcomes, BOS/PAD, WDL/FUTURE8/BOARD queries, BOARD-END and 13 piece/empty symbols. WDL query is token 1973; token 1974 means FUTURE8 despite historical “EOS” metadata. Auxiliary branches retain full history within 512-token physical rows. [1,2]

Optimization. The selected lineage uses 22,916 embedding-only adaptation updates, then full-network training with preserved optimizer/RNG state. Two-GPU NCCL/DDP uses 8 physical rows/rank, variable accumulation targeting 8,192 ordinary targets/update, BF16 autocast, FP32 parameters/gradients, deterministic masked attention and global clipping at 1. Muon updates hidden matrices: peak learning rate 4.5e-4, momentum .95, Nesterov, five Newton-Schulz steps, epsilon 1e-7, RMS-matched scaling, decay .1. AdamW handles remaining parameters: peak 3e-4, betas (.9,.95), epsilon 1e-8, decay .1 on matrices/0 on vectors. The source-progress schedule warms from 10% to peak over 5%, plateaus through 85%, then cosine-decays to 10%. Two-hour checkpoints preserve committed coverage/caps. [2]

Reproducibility identifiers

Certified corpus SHA-256: 26e9e867613e2a6ca3d8dbe939cc6de24f859898e9a2b70613b1eb02684268f3
Production manifest SHA-256: 38fd7167a43c5d8ea90b3497a35c147e627ab4c271a698af165ae656db662c78

4. Turning predictions into moves

Decode legally: normalize policy probabilities over every legal move, excluding special/outcome tokens. The live adapter uses complete standard-start history without truncation or model KV caching, and a separate fixed-White WDL softmax. The experimental model instead requires its WDL-query adapter.

Trigger lookahead: from absolute ply 3, search if the highest move probability is <.60 or its lead over second place is <.15. This measures policy concentration, not calibrated uncertainty.

Explore and score: iterative depths 1-3, top three policy moves at each node. Value is White's win probability plus half its draw probability; White maximizes and Black minimizes. This is truncated minimax. Per-decision history caching reuses evaluations; only complete root iterations replace the greedy legal choice.

Respect deadlines: root inference/search share ≤10s, further limited by a 0.5-2s reserve, approximately (remaining/30 + half increment)/active games, and queue deadline minus 150ms. An outer controller recovers over-budget workers. With no usable result, fallback is the lexicographically first legal move. [2,3]

5. Assistance and its limits

Mates. Stockfish 19 ponders up to eight opponent replies/four candidates, with ≤15s background opportunity and ≤1s position checks. Only the unused part of a 1s opportunity floor can consume the bot's clock, overlapping inference. Settings: two threads, 64 MiB hash, MultiPV 1, unrestricted strength, no fixed depth/node cap. Correctly signed, unbounded UCI mate reports justify overrides/vetoes; they are engine evidence, not formal proof certificates. Unknown evidence preserves the selected move. [3]

Draws. With supported losing-mate evidence, existing candidates are checked for immediate stalemate, insufficient material, fivefold repetition or the 75-move rule. Winning alternatives retain priority. Exact history/version/deadline checks reject stale evidence. Optional threefold/50-move claims are diagnostic only; no claims/offers are executed. Model search separately treats claimable terminals as draws. There is no general forced-draw, fortress or tablebase solver. [3]

Conversion. At ≤7 pieces, Stockfish may select its move after win probability >.97 on two consecutive own turns, requiring fresh qualifying evidence each turn. It uses one thread, 64 MiB and ≤1s within the existing deadline. Applied after mate/draw selection, it is not a permanently latched takeover. [3]

6. Evidence and next tests

Production passed 375 regressions and native-model smoke tests. Live receipts, including ep0TfgGN and its analysis, demonstrate operation, not strength. The proposed full-system evaluation uses 50 games against Stockfish and 50 against the deployed reference, each 25 color-swapped opening pairs under matched conditions. Results remain pending; earlier partial-system cohorts do not establish this system's Elo or causal gains from training changes.

For reproducibility, the verified 6 November snapshot uses CM14, release 4cee1bf258f6-a09cfbab57f0. The training recipe above is experimental CM16. Its combination with the deployed search and assistance must be tested before claiming an improvement.

References and reproducibility

[1] LittlePly. Certified corpus and preprocessing records, 6 November 2026. Original Elite, engine and 2026 additions; corpus certification, deduplication/exclusion evidence and auxiliary-task pipeline. Unpublished project records.

[2] LittlePly. Experimental training recipe and continuation lineage, 6 November 2026. Model configuration, initialization receipt, trainer and proposed full-system evaluation protocol. Unpublished project records.

[3] LittlePly. Production methods map, deployment receipt and verified 88-file release manifest, 6 November 2026. Release 4cee1bf258f6-a09cfbab57f0. Operational examples: lichess.org/ep0TfgGN; analysis.littleply.com/g/ep0TfgGN.