Diagnose, Repair, Preserve: Turn-Local Preference Reinforcement Learning for Small-Model Dialogue Game Agents

Anonymous ACL submission

Abstract

Interactive dialogue games test a capability that static benchmarks largely leave implicit: a model must carry state across turns, interpret feedback, and choose valid actions under changing constraints. We study this setting in the LM Playschool Challenge with a 2B open-weight model, and find that many failures are not broad knowledge failures but local decision failures: repeated guesses, malformed actions, and violations of feedback that the model has just seen. This motivates a training recipe organized around three steps: acquire broad game participation, focus subsequent diagnosis and turn-local repair on weak games, and preserve general capability. The resulting submitted model scores 48.70 on public validation, versus 13.05 for the 2B baseline, while slightly exceeding the baseline's static score (45.06 vs. 44.02). We also report where the recipe breaks: the same supervision can damage a stronger base model, and late preference continuation can move a sharp adapter optimum in the wrong direction.

1 Introduction

Dialogue games evaluate language models as agents whose next utterance changes the state they must later reason over. A word guess rules out letters; a clue constrains a partner's action; a navigation command changes the room from which the next command is interpreted. The clembench benchmark and the Playpen learning environment (Chalamalasetti et al., 2023; Horst et al., 2025) make these dependencies explicit: success requires not only producing plausible language, but also maintaining state, interpreting feedback, and choosing a valid action under accumulated constraints.

Playpen frames this setting as learning from Dialogue Game Feedback (Horst et al., 2025): programmatic, goal-directed feedback over trajectories, without relying on a learned judge over isolated responses. The feedback does not itself locate where an interaction stops making progress; our game-specific diagnostics do so by identifying the first mechanically repairable failure after feedback. The question for a small model is therefore not only whether dialogue game data helps, but which part of the trajectory should receive supervision.

Dialogue state	Model action	Local failure
After `crane`: red `c/r/n`; yellow `a`; green `e`	`guess: crane`	repeats guess
(same)	`guess: abid`	wrong length
(same)	`guess: cable`	uses red `c`

Table 1: Simplified Wordle failure states from our diagnostics. The model output can be syntactically close to valid while violating a constraint entailed by the immediately preceding feedback.

Small models are a natural target for this setting. They are cheaper to serve, better suited to low latency interaction, and more plausible for local assistants or assistants with limited resources, but their interactive behavior is brittle. In the LM Playschool Challenge, the official Qwen3.5-2B baseline (Qwen Team, 2026) scores only 13.05 clemscore on public validation, far below the 65.72 score of its 27B sibling.[1] The challenge also reports a static benchmark score, so a useful small model recipe must improve interaction without simply specializing away general capability.

Inspecting the 2B model's transcripts suggested that the gap was not mainly a lack of game knowledge. In Wordle, the model often preserves the expected answer format while failing at the next local decision (Table 1). These errors are frequent, mechanically checkable, and localized to a specific decision made after receiving feedback. They point to a different training question: not how to teach the model the whole dialogue again, but where to place supervision so that it reaches the particular

[1] Official public-validation baselines: https://github.com/lm-playpen/playpen/blob/main/LEADERBOARD.md.

decision that fails.

This paper describes our submission built around that question. We first use broad imitation to make the model participate in games, then diagnose the remaining failures, repair those of one weak and mechanically diagnosable game family (Wordle) with turn-local preference pairs, and finally scale the learned update to recover general capability. The submitted model reaches 48.70 clemscore, above the official 9B and 4B baselines, while slightly improving static score over the official 2B baseline (45.06 vs. 44.02).

The main contribution is an empirical account of supervision placement for small interactive agents. A small number of preference pairs anchored to real turns after feedback produces the largest useful jump in our pipeline, whereas much larger whole dialogue or teacher distillation corpora do not. The negative results are equally important: the same recipe does not transfer unchanged to a stronger 4B base, and every completed late continuation of the final adapter regresses. Together these results delimit when turn-local preference learning helps.

2 Task and Evaluation Setting

The LM Playschool Challenge evaluates whether learning in conversational interaction improves agentic capability. Playpen is the learning environment for Dialogue Game Feedback (Horst et al., 2025): it organizes rollouts, training data, and online sampling over dialogue game trajectories. The dialogue games are implemented in clembench (Chalamalasetti et al., 2023), where programmatic game masters mediate turns, track state, and enforce output protocols, while game-specific scorers compute Quality.

In the public validation setting used here (from the `playpen-data` dataset [2]), the interactive suite contains 14 clembench games and 67 episodes in our pinned run. We pin this split to revision `557d8caf`, the abbreviated Git commit identifier of the `playpen-data` snapshot resolved from its Hugging Face repository at evaluation time. The final ranking uses a private set of unseen dialogue games; public scores are iteration signals.

The LM Playschool Challenge uses Clemscore and Statscore as metrics. **Clemscore** combines two quantities for each game g: **Played**, the percentage of episodes not marked aborted by the evaluator, and **Quality**, the task score on played episodes. With $G_Q = \{g \in G : \text{Played}_g > 0\}$, we report it as

$$C = \left(\frac{1}{|G|} \sum_{g \in G} \frac{\text{Played}_g}{100} \right) \times \left(\frac{1}{|G_Q|} \sum_{g \in G_Q} \text{Quality}_g \right). \quad (1)$$

Quality is undefined for a game with no played episode, so the Quality macro-average runs over G_Q only. Thus a model can lose score either by failing to play or by playing poorly after it stays in the game. The `playpen-data` validation split contains 430 static-benchmark instances drawn from BBH (Suzgun et al., 2023), CLadder (Jin et al., 2023), EQ-Bench (Paech, 2023), IFEval (Zhou et al., 2023), and MMLU-Pro (Wang et al., 2024); applying the same aggregation yields **Statscore**.

The pinned public split contributes one or two episodes for each represented game–experiment combination, and this small sample is the main source of run-to-run variance: differences of a few points should be treated cautiously. We therefore use paired same host comparisons for development decisions and report the submitted merged artifact as the primary result.

3 Our Method

Our recipe is a narrow instantiation of the training regimes enabled by Playpen: successful transcripts support SFT, paired alternatives support DPO, and fresh rollouts support online reward learning. Playpen itself includes turn-level DPO (Horst et al., 2025); our specific choice is to use game-specific checks to construct preference pairs at the first mechanically repairable failure after feedback. We use broad SFT to make the model participate, then target those diagnosed decisions.

All training uses LoRA (Hu et al., 2022) (r=16, α=32, all-linear, dropout 0.05) on Qwen3.5-2B, with data drawn only from the `train` split of `playpen-data`. Table 2 gives the trajectory. Following the official clemscore definition in Eq. 1, broad imitation primarily improves Played; our later stages target Quality on states the model already plays; final delta scaling targets statscore rather than game behavior directly.

[2] https://huggingface.co/datasets/colab-potsdam/playpen-data.

Phase	Stage (obj.)	LR / bs / len	Clem
—	Qwen3.5-2B start	—	13.05
A. Acquire	1 broad SFT	5e-5/32/2048	43.85
	2 weak-game SFT	5e-5/32/2048	44.09
B. Probe	3 W-GRPO probe	—/256*	43.35
C. Repair	4 W-turn-DPO rpt/len	5e-6/16/1024	46.83
	5 W-turn-DPO inv/dup	5e-6/16/1024	45.52
D. Refine	6 W-branch-DPO β=.2	5e-6/16/1024	48.93
E. Preserve	delta scaling s=.85	no grad	**50.43**
	shipped fp32 merge	—	48.70

Table 2: Pipeline trajectory and per-stage setup (LR / effective batch / max length; clemscore at right, all on the pinned `557d8caf` public-validation snapshot). Stages 3–6 (marked W-) train on Wordle data only; the Clem column always reports the full 14-game suite. All training stages use LoRA on Qwen3.5-2B, AdamW, bf16, and one epoch except GRPO (two). *max new tokens; rpt/len = repeated guesses and bad length; inv/dup = invalid words and duplicated answers. Stage 6 branches the model (factor 2, rounds 1–3, T=0.7) against a frozen reference; repeated Stage 6 evaluations range 48.1–48.9. Per-game and per-stage Wordle detail: Tables 5 and 6.

3.1 Diagnosing Small Model Game Failures

The first two stages are deliberately conventional. Stage 1 performs SFT on success-only interactions balanced to $\leq$700 examples per game from the training split of `playpen-data`. Stage 2 continues only on the Stage 1 games with the lowest per-game clemscores in our validation runs: Wordle family games, Codenames, and GuessWhat. The added transcripts are success-only interactions from a manually selected shortlist of source models (GPT-5.2 families, Gemini-3-flash, Kimi-k2, Claude-Sonnet-4-5). This lifts the model from 13.05 to about 44, but then saturates. The model has learned the surface form of game play, yet raw Wordle quality remains at zero: it still repeats rejected guesses and violates immediate feedback constraints. Wordle was not the only remaining weakness, but its failures were frequent, local to a single decision, and mechanically verifiable, so we selected it as the target of all subsequent diagnosis and optimization: Stages 3–6 use only Wordle optimization data, while every reported clemscore remains the full 14-game suite.

Stage 3 tests whether online reward shaping can repair these failures directly. We run a Group Relative Policy Optimization (GRPO; Shao et al., 2024) probe with a dense Wordle reward that parses feedback and penalizes six mechanical violations, from exact repeats and bad length to letter-position constraints. Its measured marginal (43.35 versus 44.09 for its parent on the pinned snapshot; +0.36 in an earlier development evaluation on an earlier snapshot of the validation set) is well within the roughly three-point suite variation, so these evaluations cannot resolve GRPO's direct effect. The earlier result nevertheless made the GRPO checkpoint the parent of the subsequent stages, and we retain it in the submitted lineage as an online probe: its rollout and evaluation trajectories exposed persistent repeat and bad-length failures despite correct output formatting, and the failure of denser constraint-aware shaping motivated us to convert these failures into turn-local preference pairs.

3.2 Turn-Local Repair

Given a diagnosed failure mode, we construct preference pairs that share the same dialogue state. For a history h, the rejected completion y^- exhibits one mechanical error and the preferred completion y^+ is a valid alternative. We then apply Direct Preference Optimization (DPO; Rafailov et al., 2023):

$$\mathcal{L}_{\mathrm{DPO}} = -\mathbb{E} \log \sigma\Big(\beta\big[r_\theta(h, y^+) - r_\theta(h, y^-)\big]\Big), \tag{2}$$

with policy–reference log-ratio $r_\theta(h, y) = \log \frac{\pi_\theta(y|h)}{\pi_{\mathrm{ref}}(y|h)}$. Holding h fixed is the point: the contrast lands on the few decision tokens that determine whether the model repeats a guess, uses the wrong length, produces an invalid word, or duplicates the answer field. Stage 4 uses 1,905 pairs for repeated guesses and bad length, yielding the largest single jump of the project (+3.5). In the small raw Wordle probe, the old repeat loops no longer appear, but invalid words and duplicated answer fields emerge instead, motivating the Stage 5 repair batch. Its aggregate dip is within suite noise, and the resulting checkpoint becomes the immediate parent of Stage 6.

After addressing fixed mechanical failure classes with synthetic pairs, we turn to errors arising in the current model's own trajectories. Stage 6 therefore plays training split Wordle with the current model, branches at early decision points, and builds preference pairs from branch outcomes. DPO against a frozen reference (β=0.20) closes the distribution gap: the original Stage 6 evaluation reached 48.93, with repeated evaluations ranging from 48.1 to 48.9.

Model	Clem	Stat	ΔC	ΔS
Qwen3.5-27B (official)	65.72	65.99		
Qwen3.5-9B (official)	41.92	54.16		
Qwen3.5-4B (official)	39.70	50.63		
Qwen3.5-2B (official)	13.05	44.02	—	—
Varad SFT+DPO (2B)[c]	67.71	43.87	+54.7	−0.2
Varad SFT (2B)[c]	58.10	44.34	+45.1	+0.3
GuidePlay (2B)[c]	57.12	42.68	+44.1	−1.3
LLP (2B)	54.28	42.11	+41.2	−1.9
Ours (2B, submitted)	48.70	**45.06**	+35.7	**+1.0**
bosphorous (2B)	46.66	40.39	+33.6	−3.6

Table 3: Main public validation results. Deltas are against the official 2B baseline. Official baseline rows are organizer-reported; LLP and bosphorous are participant-submitted intermediate results from the organizer-hosted leaderboard snapshot dated 2026-06-16. Rows marked [c] are public model-card scores checked on 2026-07-08. Our rows are measured on the pinned `557d8caf` public-validation snapshot, with shipped merged weights verified from a clean pull.

3.3 Preserving General Capability

By this point, every strong interactive candidate pays a specialization tax: statscore drops 3–5 points from our locally re-evaluated starting checkpoint (45.79 to 40.90 at β=0.20), concentrated in EQ-Bench and IFEval while BBH improves. Rather than add more data or replay, we treat preservation as a post-training weight-space problem. We scale the learned delta, $W(s) = W_{\text{base}} + s\,\Delta_{\text{LoRA}}$, and select the scalar $s \in [0, 1]$ by constrained model selection,

$$\begin{aligned} s^\star &= \arg\max_s \; S\big(W(s)\big) \\ \text{s.t.} \quad & C\big(W(s)\big) \geq C_{\min}, \end{aligned} \tag{3}$$

where $C_{\min}$ is a paired interactive gate. At $s^\star$=0.85, statscore recovers to 44.90 while paired clem evaluation is unchanged or better (50.43 vs. a 48.11 anchor). The operation is training-free and composes with any plain LoRA adapter, but the scalar must be selected on both axes: s=0.80 already fails the interactive gate (45.22).

4 Results

Against the official 2B baseline, our shipped artifact gains +35.7 clemscore with a positive static delta. It also exceeds the official 9B and 4B baselines on clemscore.[3] Concurrent public model cards[4] report 2B systems with higher public-validation clemscores, reaching 57–68. Within this listed set of contemporaneous 2B reports, ours has the highest reported statscore among systems with higher reported clemscore (45.06 vs. 42–44).

Although Stages 3–6 use only Wordle optimization data, the final Wordle-trained checkpoint scores 48.1–48.9 on the full suite, compared with 44.09 after Stage 2, a cumulative increase of 4.0–4.8 points; the scaled artifact then scored 48.70 in the submission run and 49.10 after a clean Hub re-pull. However, most adjacent-stage differences are not clearly distinguishable from suite-level noise, and per-game changes outside Wordle are mixed (Tables 5 and 6 in Appendix), so we do not claim cross-game transfer.

The group view separates protocol robustness from conditional task quality (Table 4 in Appendix). Structured spatial communication and text-world navigation achieved 100% played rates in this run, with group clemscores of 80.1 and 57.8. Feedback-conditioned search showed high conditional quality once played, but Wordle aborts limited its played rate. Semantic clue coordination showed the opposite pattern: the model often stayed in the game, but low Codenames and Taboo quality reduced the group score to 15.6. Common-ground tracking, represented here only by PrivateShared, was primarily a play-or-abort failure: its played rate was 25.0% despite conditional quality of 73.3. This decomposition also marks the reach of our recipe: explicit local constraints in Wordle support turn-local repair, whereas clue helpfulness is reflected only indirectly in downstream game outcomes.

5 Discussion

5.1 Positive Lessons

The useful unit was the failing turn. The strongest signal in the pipeline is not a longer dialogue demonstration but a local contrast at the first bad decision. Stage 4's 1,905 turn-local pairs yield a +3.5 jump because the label is clean and the credit assignment is short: at the same history, one completion repeats a guess or has the wrong length, and the other does not.

[3] All evaluations disable Qwen3.5 thinking mode. The game master does not strip visible reasoning text; in our checks, enabling thinking caused near-total protocol failure for both the 4B and 9B baselines (clemscore 0).

[4] Model cards (accessed 8 July 2026; public validation only; all at `https://huggingface.co/`): Varad SFT+DPO, `varadsrivastava/lm-playschool-qwen3.5-2b-sft-dpo`; Varad SFT, `varadsrivastava/lm-playschool-qwen3.5-2b-sft`; GuidePlay, `syedhuq/qwen-guideplay-2b`; bosphorous, `Shamima/lm-playschool-qwen3.5-2b-sft-iter2`.

The contribution is targeted turn-local repair. Playpen already studies turn-level DPO alongside dialogue-level training (Horst et al., 2025). Our run targets the first mechanically repairable decision identified by game-specific checks: whole successful trajectories acquire the surface policy, while the highest-value later signal is a contrast between two actions at the same state. The programmatic feedback supplies verifiable outcomes; our diagnostics locate the decision to label.

Pipeline value was diagnostic rather than additive. The six-stage chain should not be read as an architecture to copy unchanged. The practical value of the pipeline was that it exposed which supervision was worth applying, not that depth itself created the gain. Our ablation study reaches the same conclusion cautiously: turn-local DPO has the clearest positive marginal, while individual stage deltas are too noisy to treat as effect sizes.

Preservation can be separated from repair. The static score loss after DPO behaves like an overlarge adapter delta rather than an irreversible loss of general capability. Scaling the LoRA delta recovers statscore without a new optimizer step, and outperformed replay mixing and learning-rate tuning in our runs — a useful property for shared task settings where specialization is rewarded but general behavior is still evaluated.

5.2 Negative Results

The recipe depends on base capability. The same supervision that lifts 2B can damage a stronger base. Applied unchanged to the 4B baseline, the recipe drops clemscore by -15.3 (39.70 to 24.36). On-policy self-play is gentler (-4.6), and a training-intensity sweep finds only one low-intensity regime that preserves 4B capability; none improves it.

Late continuation moves a sharp adapter the wrong way. After delta scaling, we tested correction-based on-policy DPO continuations. Every completed continuation scored below the 50.43 adapter candidate (best -0.7, worst -23.2), including variants using solver-generated corrections, success-versus-failure pairs, and corrections from a much stronger teacher. Every SFT continuation of the locked adapter drove Wordle played rate to zero; the useful operation was to rescale the delta, not to keep taking gradient steps.

More data was not better credit assignment. A 15,490-row teacher distillation corpus with heavy anchor replay loses 10.8 points. Whole dialogue preference tuning on weak games was abandoned, and isolated subskill training failed to compose. These failures are the mirror image of the positive result: a verifiable local decision can be more valuable than thousands of tokens of correct but diffuse demonstration.

6 Conclusion

A 2B model's interactive game deficit can be a local decision problem rather than a broad knowledge problem. In that regime, supervision is most effective at the failing turn, not spread across an entire dialogue. Our submission uses that principle to lift the weakest official baseline above the larger official baselines on public interactive score while preserving static performance. The same experiments mark the boundaries: the recipe does not transfer unchanged to stronger bases, and late continuation can degrade a sharp adapter.

Limitations

Public clem validation is noisy (67 episodes in our verified run; $\sim$3 point variance), so despite paired same host comparisons our interactive numbers carry real uncertainty and the private set ranking may track them poorly. The recipe is shown on one base and one game family and does not transfer upward unchanged (§5.2); the statscore recovery is a post hoc intervention in weight space, not loss-free training; and stage-level attributions are unreliable at this suite size, so the phase decomposition should not be read as a validated additive one.

Reproducibility Statement

For anonymous review we withhold identifying repository, model-card, and training-card links; these are restored camera-ready with exact commands, per-stage configs, and data-generation scripts.[5]

References

Kranti Chalamalasetti, Jana Götze, Sherzod Hakimov, Brielen Madureira, Philipp Sadler, and David Schlangen. 2023. clembench: Using game play to

[5]Repository, model-card, and training-card links withheld for anonymous review; restored in the final version.

evaluate chat-optimized language models as conversational agents. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 11174–11219, Singapore. Association for Computational Linguistics.

Nicola Horst, Davide Mazzaccara, Antonia Schmidt, Michael Sullivan, Filippo Momentè, Luca Franceschetti, Philipp Sadler, Sherzod Hakimov, Alberto Testoni, Raffaella Bernardi, Raquel Fernández, Alexander Koller, Oliver Lemon, David Schlangen, Mario Giulianelli, and Alessandro Suglia. 2025. Playpen: An environment for exploring learning from dialogue game feedback. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 29854–29891, Suzhou, China. Association for Computational Linguistics.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Zhijing Jin, Yuen Chen, Felix Leeb, Luigi Gresele, Ojasv Kamal, Zhiheng Lyu, Kevin Blin, Fernando Gonzalez Adauto, Max Kleiman-Weiner, Mrinmaya Sachan, and Bernhard Schölkopf. 2023. CLadder: Assessing causal reasoning in language models. In *Advances in Neural Information Processing Systems*, volume 36, pages 31038–31065. Curran Associates, Inc.

Samuel J. Paech. 2023. EQ-Bench: An emotional intelligence benchmark for large language models. *Preprint*, arXiv:2312.06281.

Qwen Team. 2026. Qwen3.5: Towards native multimodal agents.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, volume 36, pages 53728–53741. Curran Associates, Inc.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *Preprint*, arXiv:2402.03300.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc Le, Ed Chi, Denny Zhou, and Jason Wei. 2023. Challenging BIG-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13003–13051, Toronto, Canada. Association for Computational Linguistics.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. 2024. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. In *Advances in Neural Information Processing Systems*, volume 37, pages 95266–95290. Curran Associates, Inc.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. 2023. Instruction-following evaluation for large language models. *Preprint*, arXiv:2311.07911.

Analysis group	Games	Pl.	Qu.	Clem
Feedback-cond. search	Wordle × 3, GuessWhat	58.3	75.0	43.8
Semantic clue coord.	Codenames, Taboo	69.2	22.5	15.6
Spatial communication	ReferenceGame, ImageGame, MatchIt ASCII	100.0	80.1	80.1
Common-ground tracking	PrivateShared	25.0	73.3	18.3
Text-world navigation	AdventureGame, TextMap-World × 3	100.0	57.8	57.8

Table 4: Analysis-group decomposition from a clean Hub re-pull of the submitted fp32 artifact (overall clem 49.10; the 48.70 in Table 3 is an earlier independent run of the same artifact, within suite noise). Groups are our descriptive partition by dominant interaction demand, not an official clembench taxonomy. Pl. is mean % played; Qu. is mean conditional quality over games with played episodes; Clem = Pl. × Qu. / 100. The common-ground group contains only PrivateShared. Wordle's zero played rate in this run is analyzed in Table 6.

A Analysis-Group Decomposition

Table 4 groups the 14 games by dominant interaction demand and is discussed in §4.

B Per-Game Matched Comparison

Table 5 decomposes the pipeline per game at four checkpoints: the vanilla base, the last cross-game checkpoint (Stage 2), the last Wordle-trained checkpoint (Stage 6, before delta scaling), and the shipped fp32 merge. To make the per-game deltas internally comparable, all four checkpoints were re-evaluated on a single host with the same evaluation harness, dataset revision `557d8caf`, and clembench commit `ed39486`; the matched base score coincides with the organizer-reported baseline (13.05). These matched runs support the per-game analysis only; the official numbers in the body (the 13.05 organizer baseline and the 48.70 submitted score) are unchanged.

The three delta columns separate what changed during cross-game SFT (Base→S2), during the Wordle-only stages (S2→S6), and under gradient-free delta scaling plus the fp32 merge (S6→Final). Changes outside Wordle during the Wordle-only stages are mixed in sign and rest on 3–13 episodes per game; we report them descriptively and do not claim cross-game transfer.

C Plain-Wordle Behavior by Stage

Table 6 tracks the three public plain-Wordle episodes across the pipeline, using process-level metrics rather than the game score alone. Among checkpoints with played plain-Wordle episodes, mean repetitions fall from 3.0 at Stages 1–3 to 0 at Stage 5; repetition is unobservable at Stage 4 and after scaling because no episode is played. Stage 6 shows the most stable protocol behavior of the lineage (100% played, no violated requests, 100% request success), yet conditional Quality is 0 at every checkpoint with played episodes: the Wordle-focused stages repaired protocol-level failures on this public probe but did not solve the underlying guessing task.

Across repeated same-harness evaluations, the unscaled Stage-6 checkpoint plays all three public episodes, while the s=0.85-scaled checkpoint and all downstream artifacts abort all three, re-introducing a bad-length guess of the kind targeted in Stage 4. We report this as a reproducible observation on the fixed three-episode probe rather than an independently estimated causal effect of delta scaling.

Game	n	Base	Stage 2	Stage 6	Final	Δ B→S2	Δ S2→S6	Δ S6→F
AdventureGame	5	0.0	33.3	40.0	33.3	+33.3	+6.7	−6.7
Codenames	13	0.0	0.0	7.7	7.7	0.0	+7.7	0.0
GuessWhat	6	0.0	50.0	33.3	50.0	+50.0	−16.7	+16.7
ImageGame	4	35.8	81.2	89.8	85.2	+45.5	+8.5	−4.5
MatchIt ASCII	4	75.0	50.0	75.0	75.0	−25.0	+25.0	0.0
PrivateShared	4	0.0	35.1	21.3	18.3	+35.1	−13.8	−2.9
ReferenceGame	5	100.0	80.0	60.0	80.0	−20.0	−20.0	+20.0
Taboo	6	0.0	41.7	41.7	25.0	+41.7	0.0	−16.7
TextMapWorld	5	0.0	55.6	65.2	65.7	+55.6	+9.6	+0.5
TMW GraphReasoning	3	0.0	58.9	58.9	58.9	+58.9	0.0	0.0
TMW SpecificRoom	3	0.0	66.7	66.7	66.7	+66.7	0.0	0.0
Wordle	3	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Wordle w/ Clue	3	0.0	33.3	50.0	50.0	+33.3	+16.7	0.0
Wordle w/ Critic	3	0.0	66.7	66.7	66.7	+66.7	0.0	0.0
Overall	67	**13.05**	**43.85**	**48.52**	**48.70**	+30.80	+4.67	+0.18

Table 5: Matched per-game comparison of four checkpoints (single host, single harness, pinned dataset `557d8caf` and clembench `ed39486`). The per-game score is the derived $P_g Q_g/100$; it is displayed as 0 when Played $= 0$, although Quality itself is undefined there (Eq. 1). The overall clemscore is the product of macro-averaged Played and macro-averaged Quality over played games, not the mean of the per-game column, so per-game deltas do not sum to the overall delta. Stage 6 here is a matched re-evaluation of the checkpoint whose original evaluation reached 48.93 (Table 2). Δ S6→F includes gradient-free delta scaling and the fp32 merge, not additional training. Each game has 3–13 episodes; the changes are descriptive.

Checkpoint	Training scope	Suite Clem	W Played	W Quality	Rep./played ep.	Viol./ep.	Req. success
Stage 1	all games	43.85	66.7	0	3.00	0.33	88.9
Stage 2	weak games	44.09	66.7	0	3.00	0.33	88.9
Stage 3	Wordle	43.35	66.7	0	3.00	0.33	88.9
Stage 4	Wordle	46.83	0	—	—	1.00	63.9
Stage 5	Wordle	45.52	66.7	0	0.00	0.33	91.7
Stage 6	Wordle	48.93; 48.11×2	100	0	0.00	0.00	100.0
+ scaling s=.85	none (no gradient)	50.82 / 50.03	0	—	—	1.00	69.4
Shipped fp32 merge	none (merge only)	48.70 / 49.10	0	—	—	1.00	69.4

Table 6: Plain-Wordle behavior after each stage on the three public validation episodes (development evaluations; pinned dataset `557d8caf`). Suite Clem is the full 14-game clemscore; for Stage 6 we list the original evaluation and two paired re-evaluations, and for the scaled and shipped artifacts the two runs of each. W Quality is undefined (—) when no episode is played. Rep./played ep. is the mean clembench *Guess Repetitions* over played episodes; Viol./ep. and Req. success are the mean *Violated Request Count* and *Request Success Ratio* (%) over all three episodes.