Teaching or Sharpening? An Exploration of the Potential of DPO Post-Training for Dialogue Games

Anonymous ACL submission

Abstract

Direct Preference Optimization (DPO) has emerged as a lightweight alternative to RLHF for post-training LLMs, yet it remains unclear whether it can teach novel skills or only sharpen existing ones. We investigate this question in the context of dialogue games, constructing multiple sets of on-policy preference pairs from the rollouts of supervised-fine-tuned Qwen3.5 models (2B and 9B) on the clembench benchmark, targeting rule adherence, strategic behaviour and excessive verbosity. Across all ablations, performance tracks a single variable, namely the distance of the chosen response from the policy's own distribution. Pairs built from the model's own winning moves raise the clemscore from 58.82 to 70.21 while pairs built around GPT-corrected moves land below the baseline at both scales. Transcript analysis points to a structural reason: bare preference pairs reduce conditional rules to unconditional preferences, so that DPO proves able to elicit successful behaviour already latent in the policy, but not to instill behaviour it does not yet exhibit.

1 Introduction

Dialogue Games, defined as "constructed activities driven by language use" (Schlangen, 2023), have been gathering increasing interest among the Natural Language Generation and Understanding research communities, as they can be used as gyms to train and test LLM-based agents' interactive reasoning capabilities (Chalamalasetti et al., 2023; Horst et al., 2025).

Experimental findings suggest that passive exposure to text via self-supervised next-token-prediction over static corpora might not be a powerful enough learning signal to train models towards handling language in dialogue (Horst et al., 2025; Bisk et al., 2020; Laban et al., 2026; Sarzotti et al., 2025), and therefore various attempts have been carried out to explore the potential of interaction-based (post-)training. Reinforcement-based approaches in particular have been tested towards this objective, as their "trial-feedback-error" pipelines appear to model the dynamicity of learning in interaction better than more static training regimes such as Supervised Fine-Tuning (Horst et al., 2025; Carta et al., 2023; Cheng et al., 2024). However, a commonly understood limitation of online reinforcement learning techniques is their cost in terms of computation and resources. Direct Preference Optimization (Rafailov et al., 2023) offers a valid alternative specifically to preference-based RL pipelines such as RLHF by approximating preference optimization to a binary contrastive classification problem.

In this landscape, with the present work we aim to test the potential of DPO as a post-training approach in the specific context of dialogue games, in particular asking if a debate originally developed around reinforcement learning — namely, whether such methods are able to teach novel skills to the model (Chu et al., 2025; Liu et al., 2025) or whether they are better suited for sharpening competences already acquired through previous stages (Huang et al., 2025; Gandhi et al., 2025; Yue et al., 2025; Zhao et al., 2025) — extends to DPO as well. To this end, multiple sets of on-policy preference pairs for training were constructed starting from the supervised-fine-tuned models' own rollouts on the full clembench benchmark (Chalamalasetti et al., 2023), incrementally tackling combinations of three behaviours that can significantly impact performance in dialogue games, namely the more fine-grained skills of rule-following and strategic thinking, along with general verbosity-driven rambling and looping. Ablation studies revealed that the best performing model was the one solely optimized against excessive verbosity, which increased the benchmark-specific clemscore from 58.82 to 70.21 w.r.t. the supervised-fine-tuned baseline, nudging our findings towards the claim that

preference-based post-training approaches are the most helpful when it comes to sharpening already acquired tendencies. We thus present a technical breakdown of our best model — submitted to the LM Playschool Challenge — alongside an evaluation of how the different training ablations impacted the behaviour and performance of models of two varying sizes, precisely Qwen3.5-2B and Qwen3.5-9B (Qwen Team, 2026).

2 Methods

2.1 Base Model and Adaptation

As starting points for the present study we used a Qwen3.5-9B model and Qwen3.5-2B model. Both are instruction-tuned reasoning models, which we loaded in non-thinking mode to avoid an explosion in compute and wall-clock time at inference. We chose to use models of the Qwen family because they rank among the top open-weight models on the clembench benchmark [1] (Chalamalasetti et al., 2023), and specifically selected the two above-mentioned models for ablation comparison, to analyze how the impact of the proposed methodology scales with number of parameters.

The models were trained following a two stage pipeline, which involved a first phase of supervised fine-tuning training and a subsequent DPO one.

SFT was conducted by training LoRA adapters (Hu et al., 2022) on the successful game rounds in the playpen dataset [2] (Horst et al., 2025) via assistant-only language-modeling loss. The learnt adapters were applied to all linear layers of the base models and merged with them to obtain ready-made models to use in the following stage.

DPO fine-tuning made use of on-policy preference pairs constructed starting from the supervised-fine-tuned models' own rollouts on the full clembench benchmark. Apart from data curation, all the other training details remain constant across conditions. During the DPO stage, LoRA adapters were trained with the standard sigmoid loss (Rafailov et al., 2023). All training hyperparameters are reported in Table 1. Full replication instructions for the best model are provided in our GitHub repository.

The submitted model was trained on a single NVIDIA H100 (94 GB), with SFT taking 3 h 50 m of wall-clock time and DPO a further 25 m. The full ablation study — 2 SFT runs, 2 rollout harvests, 14 DPO runs, and 18 evaluated checkpoints — ran as single-GPU jobs on NVIDIA A100 80 GB cards and one H100 94GB, for an estimated total of 25–35 GPU-hours.

Hyperparameter	SFT	DPO
Optimizer	AdamW	AdamW
Epochs	1	1
Learning rate	2e-4	5e-5
Effective batch size	16	8
LoRA rank / α	16 / 32	16 / 32
β	0.3	
Max sequence length	1024	2048

Table 1: Training hyperparameters for the SFT and DPO stages. All runs used seed 42.

2.2 Dataset

The SFT training phase was conducted using the successful game rounds in the interactions subset of the playpen dataset's train split. Only the game rounds that were carried out successfully — i.e. resulting in the player model winning — were retained for training, for a total of 20,202 game rounds. For the DPO stage, on the other hand, we started by collecting on-policy rollouts of the supervised-fine-tuned models on the whole clembench v2.0 benchmark. Rounds that are part of the validation subset of the playpen dataset — and that will therefore be used later to evaluate the models' performances — were excluded. Such data was then curated in specific ways to construct preference pairs for three main training ablations, each one tackling a different skill or behaviour that can easily impact game-playing performance: adherence to formatting rules, strategic behaviour and excessive verbosity.

2.2.1 Rule adherence

In order to construct the dataset of preference pairs targeting rule adherence, the rollouts which ended in the game rounds being aborted were extracted. In the context of clembench, a round is interrupted and marked as aborted when the player model fails, in its output answer, to respect game-specific formatting rules.

For each on-policy rollout which encountered this outcome, a GPT-5.2 model was prompted to generate a better, rule following answer in place of the faulty one provided by the SFT baseline model.

The dataset of preference pairs was then constructed by using the whole game rounds' conver-

[1] Consult the clembench leaderboard and the full list of games in the benchmark on the dedicated website.

[2] Dataset available on HuggingFace

sation history up until the rule breaking answer as prompt, the GPT-corrected move as "chosen" (positive) continuation and the one originally proposed by the player model as "rejected" (negative) continuation. We thus obtained a dataset comprising 42 pairs for the Qwen3.5-9B SFT model and 79 for the Qwen3.5-2B SFT one.

2.2.2 Strategic Behaviour

Preference pairs tackling strategic game-playing were constructed in a very similar way to those described above for rule adherence. This time, however, the on-policy rollouts used were those where game rounds were failed (completed without rule violations, but not won).

The same GPT-5.2 model was prompted to identify, among the conversation history of each failed round, the first instance where the SFT model made a strategically poor move, and suggest a better one. Navigation-based partially observable games — namely games in the textmapworld family and adventuregame — were passed to GPT separately from all the others and under a different prompt. Given that the GPT model has access to the whole conversation history, it was necessary to clearly state in its prompt that any better move suggested had to be based solely on the information that could have been available to the player at that specific time in the game. We obtained 286 pairs for the 9B model and 325 for the 2B one.

2.2.3 Anti-verbosity

Finally, we constructed a set of preference pairs to tackle the generic tendency of LLMs to produce overly-verbose text and ramble in their outputs.

After extracting the successful rollouts of the SFT model, we sampled up to two assistant messages each and constructed the preference pairs with the player model's own move as "chosen" and, as "rejected", the same move with one of five hand-written verbose continuations of empty prose appended to it. We thus obtained 995 pairs for the 9B model and 770 pairs for the 2B one.

Dataset imbalances are due to the availability of suitable rollouts being strictly dependent on how many rounds played by the two models fell under each of the three outcomes.

3 Results

All conditions are evaluated on the same 68 clembench instances of the playpen validation split, decoded at temperature zero, so all comparisons in this section are paired per instance. We report clemscore, the harmonized clembench metric obtained as the product of %Played, which measures the share of episodes completed without formatting violations, and Quality, the average game-specific score on completed episodes. Alongside it we report statscore, the analogous score on playpen's static benchmark suite, as a check on general capabilities [3]. Throughout this section we abbreviate the three conditions of Section 2.2 as AB (rule adherence, built from aborted rounds), F (strategic

	9B		2B	
Condition	**clem**	**stat**	**clem**	**stat**
Qwen3.5 (no fine-tuning)	41.92	54.16	13.05	44.02
SFT baseline	58.82	61.09	38.05	42.88
Anti-verbosity (AV)	**70.21**	58.73	44.69	37.38
Chosen-only (ctrl.)	65.36	59.35	46.41	44.82
Rule adher. (AB)	60.96	56.82	**54.28**	42.11
Strategy (F)	44.43	61.55	25.26	42.59
AV+AB	63.69	60.28	42.50	40.79
AB+F	43.45	61.58	28.81	39.81
AV+F	42.71	58.54	24.94	39.57
All three	26.92	43.55	29.45	33.02

Table 2: clemscore and statscore on the playpen validation split for the 9B and 2B models. The first two rows are reference points, namely the untuned Qwen3.5 models and the SFT baselines from which all conditions start. Following rows are ordered by the distance of the chosen response from the policy's own distribution: own winning move (AV, chosen-only), minimally edited own move (AB), GPT-written move (F).

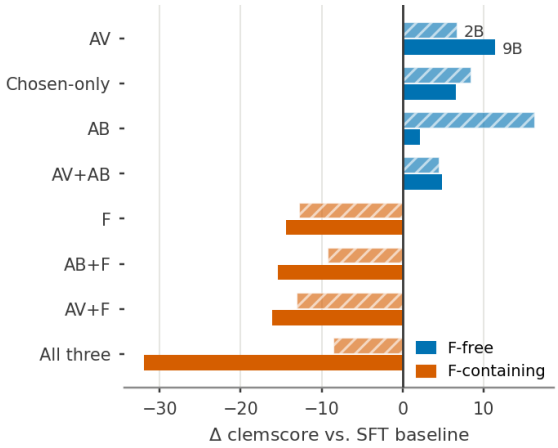

Figure 1: Clemscore difference vs. each scale's SFT baseline for all conditions, ordered by the distance of the chosen response from the policy (hatched bars = 2B).

[3] Playpen's static benchmark suite contains 430 instances sampled across an array of five "static" — i.e. not involving dialogue and interaction — conventional NLP benchmarks, namely BBH, CLadder, EQ-Bench, IFEval and MMLU-Pro.

behaviour, built from failed rounds) and AV (anti-verbosity).

As a first reference point, Table 2 reports the results obtained by the untuned 2B and 9B models. The SFT training stage alone provides the largest single improvement observed in this study — $+16.9$ clemscore at 9B and $+25.0$ at 2B over the untuned models — and the resulting supervised-fine-tuned models serve as the baseline against which all DPO conditions are compared throughout this section.

As Table 2 shows, performance decreases monotonically with the distance of the chosen (positive) response in the preference pairs from the policy's own distribution (Figure 1). Considering the 9B models' clemscore differences over the SFT baseline, the model's own winning moves (AV) yield $+11.4$, minimally edited own moves (AB) $+2.1$, GPT-written moves (F) -14.4, and every condition containing the F pairs lands below the baseline at both scales. Statscores, on the other hand, were largely preserved across conditions and sizes, with only the three-way mixture showing substantial degradation, suggesting that game-targeted training left general capabilities mostly untouched.

Mixtures of conditions consistently underperform the sum of their parts, as additivity would predict $+13.5$ for AV+AB where $+4.9$ is observed, and -0.9 for the three-way mixture where -31.9 is observed — a gap of 8.7 and 31.0 points respectively. This interference pattern generalizes across scales, as at 2B combining the two individually best conditions (AB $+16.2$, AV $+6.6$) yields $+4.5$ against an additive prediction of $+22.9$, leaving AV+AB below both of its ingredients. Full table and visualization in Appendix A. Finally, while the best performing model flips with scale, with AV first at 9B and AB first at 2B, the structure of the grid is preserved exactly, as every F-free condition improves on the baseline, every F-containing one lands below it, and the chosen-only control ranks second at both scales.

Against the baseline, AV at 9B gained 17 wins and lost 2 (net $+15$, exact McNemar $p = 0.0007$), which makes it the only condition with a statistically significant gain. 15 of the 17 gained wins are conversions of failed episodes into successes, the number of aborted episodes is unchanged, and the entire lift accordingly sits in Quality (+12.0) rather than %Played (+1.0). Notably, the verbosity that the AV pairs were designed to suppress was not reduced, as the model produces 14.7 words per move against the baseline's 12.5, with a median paired delta of exactly zero, and its preambles and format-violation rates are identical to the baseline, while Quality on played episodes rose by 15 points (Wilcoxon $p = 0.0014$). This dissociation between unchanged length and improved play replicates at 2B (Quality +15.5, $p = 0.005$).

The chosen-only control recovers most of the gain at 9B (65.36, net $+8$ wins) and all of it at 2B, where it surpasses AV itself (46.41 vs. 44.69), though not the scale's best condition, namely AB.

Gains and damage are localized by game structure. Word games with strict answer formats — namely the wordle family and taboo — respond best to AB at both scales, reference games — namely, referencegame, imagegame and matchit — are largely insensitive to every condition, and dialogue-heavier games — guesswhat, codenames and privateshared — being the hardest group at both scales, are reliably improved by no manipulation. Exploration games, namely the navigation-based partially observable games listed in Section 2.2, carry most of the variance in the grid, as they gain $+28$ to $+35$ under every F-free condition and drop to nearly zero under every F-containing one, even though most of the strategy pairs were harvested from these very games. The AV gain in this group is clearly behavioural and driven by persistence, since at 9B the SFT model quits almost immediately (a mean of 2.0 moves per episode in textmapworld, typically a single move followed by `DONE`), whereas the AV model plays an average of 8.8 moves per episode and converts all five textmapworld instances into wins. The failure of the F pairs with respect to this kind of game has a legible mechanism that drags performance in the opposite direction to that of the AV pairs. The most frequent correction in the F condition rejects a premature `DONE`, but since the pairs carry only bare moves (Section 2.2), no rationale is included and `DONE` never appears as chosen, so that the pairs uniformly penalize stopping — not only when triggered too soon in the round — and never reward it. In the textmapworld family, the F model produces `DONE` once in the whole set (0.5% of moves). It explores 19 moves per episode and aborts 10 out of 11 episodes at the turn limit, even though its moves remain perfectly well-formed. At 2B the effect is total, with zero `DONE` across 220 moves and 11 out of 11 episodes aborted. By contrast, the AV model, trained on pairs containing chosen moves drawn from complete winning episodes — and therefore including well-timed final `DONE`s — uses the command in 13.9% of its moves and

solves all 11 episodes. Per-game breakdowns are reported in Appendix B.

4 Discussion and Conclusion

Our work and findings can be positioned within the debate on the potential of reinforcement- and preference-based post-training approaches, and whether they are more suited for sharpening capabilities already encoded in the model's own distribution (Huang et al., 2025; Gandhi et al., 2025; Yue et al., 2025; Zhao et al., 2025) or to teach new ones (Chu et al., 2025; Liu et al., 2025). We call *elicitation* the raising of the probability of behaviour the model already produces, and *instillation* the attempt to inject behaviour it does not. Our results (Table 2) are explained by a single variable, namely how far the chosen response sits from the policy's own distribution. This aligns with the difficulties reported by Horst et al. (2025), whose DPO conditions — built from off-policy pairs drawn from a static corpus of other models' interactions — failed to improve over their SFT baseline, and extends the known advantage of on-policy preference data (Tajwar et al., 2024) to a graded phenomenon, as in our grid performance degrades with every step the chosen response takes away from the policy's own outputs. The AV pairs represent pure elicitation, as both sides of each pair share almost identical content. Moreover, the appended padding is text the model essentially never generates, and the 9B model is not, on its own, verbose enough for it to affect game-playing. The extra text in the rejected moves thus does not act as a suppressing signal so much as a signal to produce more of the model's own successful behaviour. The gain is quality-driven and length-neutral, it largely survives without the preference contrast — as the chosen-only control shows — and it amplifies whatever behaviour each game's winning rounds happen to instantiate, such as persistence in exploration or safe guessing in codenames. Where the SFT model never won in the harvested rollouts, as in plain wordle, no pairs exist to display proficient behaviour, and so nothing can be elicited and nothing moves performance-wise.

Instillation, on the other hand, did not merely fail, as every F-containing condition landed below the SFT baseline at both scales, and this outcome has a structural reason. Such motivation cannot be attributed to corrections' quality, as GPT's suggested moves verify as overwhelmingly legal and feedback-consistent against game ground truth (Appendix C). Rather, a correction such as "stopping is wrong while unvisited exits remain" expresses a conditional rule, but preference pairs composed of bare moves have no channel for such rationale — as they can directly express *what* was preferred but not *why* — other than the indirect information provided by the round's chat history, which encodes a signal about *when* a specific choice should be preferred, but was not strong enough as to carry information regarding the underlying reason. The model thus learnt an unconditional rather than context-dependent preference in navigation-based games at both scales, namely that "stopping is bad". The resulting damage is thus a corrupted decision rather than a malformed output. Consistent with this, F's statscore is preserved, indicating a bad game-specific strategy rather than degraded general capabilities. This instillation-oriented regime is also the one in which DPO's gradients act at their most destructive, since they simultaneously suppress a high-probability behaviour of the model and inflate text it cannot yet produce, the setting where likelihood displacement is known to arise (Razin et al., 2025).

The fact that the best-performing condition changes with model size follows from the same account, since which pair type increases scores the most depends on where the model's headroom lies. The 9B rarely breaks formatting rules, so its headroom is in the quality of play and self-imitation wins, while the 2B aborts frequently, so minimally edited corrections win — although even there most of the gained wins are converted losses rather than rescued aborts. Both scales therefore improve in the same way, namely by playing better on the episodes they already play to completion.

Returning to the question in our title, our findings show that DPO was not able to teach any of the three skills it was aimed at, corroborating the idea that such method is not the best suited for instilling novel capabilities into a model's distribution. On the other hand, the preference-based training was able to elicit, wholesale, the successful behaviour latent in the policy, sharpening the models' already acquired skills in handling different game types, and bringing substantial gains in the final scores. The best model in this study thus came from letting the model be its own teacher, with DPO functioning as an amplifier on the policy's own successes, and as a liability the moment it was pointed at anything else.

Limitations

Evaluation uses 68 validation instances at temperature zero. Paired tests are exact, but per-game cells hold 3 to 13 episodes, and powering per-game claims would require running the models on the whole clembench benchmark. For this work specifically, our compute budget prioritized the breadth of the ablations grid over replication depth. Therefore, each condition is a single training run, and multi-seed replication of the ablations would be needed to gain statistical soundness. Temperature zero decoding leaves no within-condition variance. For future work, sampled evaluation would be necessary to conduct pass@k-style tests of whether the amplified wins already exist in the SFT policy's samples. Significance is reported uncorrected. The AV result survives Bonferroni correction, the secondary ones do not. Results cover one model family, and the chosen-only control exists only for the AV condition, leaving the dispensability of the preference contrast untested for the other sets of pairs. Corrections were produced by a single judge model, and our validation certifies their legality and feedback-consistency, but not their strategic optimality.

References

Yonatan Bisk, Ari Holtzman, Jesse Thomason, Jacob Andreas, Yoshua Bengio, Joyce Chai, Mirella Lapata, Angeliki Lazaridou, Jonathan May, Aleksandr Nisnevich, Nicolas Pinto, and Joseph Turian. 2020. Experience Grounds Language. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8718–8735, Online. Association for Computational Linguistics.

Thomas Carta, Clément Romac, Thomas Wolf, Sylvain Lamprier, Olivier Sigaud, and Pierre-Yves Oudeyer. 2023. Grounding Large Language Models in Interactive Environments with Online Reinforcement Learning. In *Proceedings of the 40th International Conference on Machine Learning*, pages 3676–3713. PMLR.

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

Pengyu Cheng, Yong Dai, Tianhao Hu, Han Xu, Zhisong Zhang, Lei Han, Nan Du, and Xiaolong Li. 2024. Self-playing adversarial language game enhances LLM reasoning. In *Advances in Neural Information Processing Systems*, volume 37, pages 126515–126543. Curran Associates, Inc.

Tianzhe Chu, Yuexiang Zhai, Jihan Yang, Shengbang Tong, Saining Xie, Dale Schuurmans, Quoc V Le, Sergey Levine, and Yi Ma. 2025. SFT memorizes, RL generalizes: A comparative study of foundation model post-training. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 10818–10838. PMLR.

Kanishk Gandhi, Ayush K Chakravarthy, Anikait Singh, Nathan Lile, and Noah Goodman. 2025. Cognitive behaviors that enable self-improving reasoners, or, four habits of highly effective STars. In *Second Conference on Language Modeling*.

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*.

Audrey Huang, Adam Block, Dylan J Foster, Dhruv Rohatgi, Cyril Zhang, Max Simchowitz, Jordan T. Ash, and Akshay Krishnamurthy. 2025. Self-improvement in language models: The sharpening mechanism. In *The Thirteenth International Conference on Learning Representations*.

Philippe Laban, Hiroaki Hayashi, Yingbo Zhou, and Jennifer Neville. 2026. LLMs get lost in multi-turn conversation. In *The Fourteenth International Conference on Learning Representations*.

Mingjie Liu, Shizhe Diao, Ximing Lu, Jian Hu, Xin Dong, Yejin Choi, Jan Kautz, and Yi Dong. 2025. ProRL: Prolonged reinforcement learning expands reasoning boundaries in large language models. In *Advances in Neural Information Processing Systems*, volume 38, pages 17998–18031. Curran Associates, Inc.

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.

Noam Razin, Sadhika Malladi, Adithya Bhaskar, Danqi Chen, Sanjeev Arora, and Boris Hanin. 2025. Unintentional unalignment: Likelihood displacement in direct preference optimization. In *The Thirteenth International Conference on Learning Representations*.

Marika Sarzotti, Giovanni Duca, Chris Madge, Raffaella Bernardi, and Massimo Poesio. 2025. MLLMs Construction Company: Investigating Multimodal LLMs' Communicative Skills in a Collaborative Building Task. In *Proceedings of the Eleventh Italian Conference on Computational Linguistics (CLiC-it 2025)*, pages 1037–1047, Cagliari, Italy. CEUR Workshop Proceedings.

David Schlangen. 2023. Dialogue Games for Benchmarking Language Understanding: Motivation, Taxonomy, Strategy. *arXiv preprint*. ArXiv:2304.07007 [cs.CL].

Fahim Tajwar, Anikait Singh, Archit Sharma, Rafael Rafailov, Jeff Schneider, Tengyang Xie, Stefano Ermon, Chelsea Finn, and Aviral Kumar. 2024. Preference fine-tuning of LLMs should leverage suboptimal, on-policy data. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 47441–47474. PMLR.

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Yang Yue, Shiji Song, and Gao Huang. 2025. Does reinforcement learning really incentivize reasoning capacity in LLMs beyond the base model? In *Advances in Neural Information Processing Systems*, volume 38, pages 57654–57689. Curran Associates, Inc.

Rosie Zhao, Alexandru Meterez, Sham M. Kakade, Cengiz Pehlevan, Samy Jelassi, and Eran Malach. 2025. Echo chamber: RL post-training amplifies behaviors learned in pretraining. In *Second Conference on Language Modeling*.

Appendix

In the following, we report additional details regarding the present work. Specifically, Section A shows the full table with clemscore Δs as predicted by additivity and the ones we actually obtained, along with a visualization, Section B presents a per-game breakdown of our results and Section C outlines further information on how the GPT corrections were validated.

A Additivity Table

Table 3 reports, for every pair-type mixture at both scales, the observed clemscore Δs against the additive prediction obtained by summing Δs of the single condition constituents, while Figure 2 visualizes the same relationship. The gap is negative in all eight cases, and at 2B two mixtures with positive additive predictions (AB+F and All three) still land below the baseline, showing that interference can reverse the sign of predicted benefits rather than merely shrink them.

Scale	Mixture	Predicted	Observed	Gap
9B	AV+AB	+13.5	+4.9	−8.7
9B	AB+F	−12.3	−15.4	−3.1
9B	AV+F	−3.0	−16.1	−13.1
9B	All three	−0.9	−31.9	−31.0
2B	AV+AB	+22.9	+4.5	−18.4
2B	AB+F	+3.4	−9.2	−12.7
2B	AV+F	−6.2	−13.1	−7.0
2B	All three	+10.1	−8.6	−18.7

Table 3: Observed clemscore Δs vs. the SFT baseline for every pair-type mixture, against the additive prediction (sum of the constituent single conditions' Δs). The gap (observed − predicted) is negative in all eight cases. At 2B, two mixtures (AB+F, all three) have positive additive predictions yet land below the baseline.

B Per-Game Breakdown of Results

Figures 3 and 4 show heatmaps of the distributions of clemscore Δs with respect to the SFT baselines for the 2B and 9B models respectively. It is possible to appreciate how, across sizes, overall performance of the post-trained models decreases the moment F-pairs enter the training data. As stated in Section 3, for the 2B model most of the gains were observed in the AB condition, while the training that the 9B one benefitted from the most was AV. Moreover, in the case of the 9B model, the heatmap shows how the games that post-training (except for when F-pairs were introduced) helped the most were the navigation-based ones, then, less

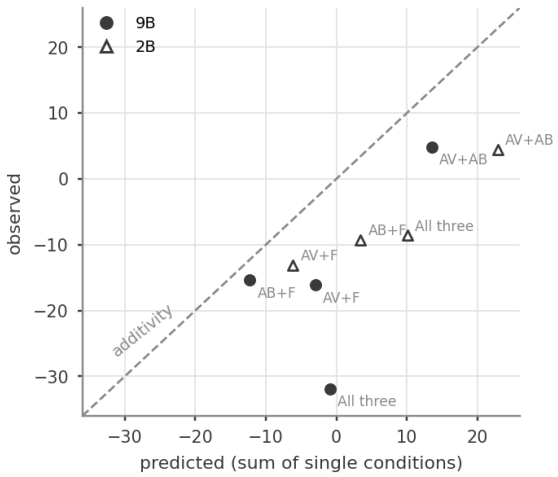

Figure 2: Observed clemscore Δs (vs. the SFT baseline) against the additive prediction for every pair-type mixture (filled circles: 9B; open triangles: 2B). Points on the dashed identity line would indicate that pair types combine additively. All eight mixtures fall below it.

consistently, the text games with strict formatting rules, then reference games — where hardly any gain is observed, potentially because performance was already very high, even in the untuned model [4] — and finally dialogue-heavier games, arguably the hardest family since performance was steadily below that of the other kinds of games, even under the most beneficial ablations.

C Validation of GPT-Generated Corrections

As outlined in Section 2.2, the chosen (positive) responses in AB and F preference pairs for DPO training consist of bare corrected moves generated by a GPT-5.2 model.

In order to test the potential presence of confounds driven by poor correction quality — especially as a possible cause for the F conditions' damage to task performance — we validated every correction in both pair sets and at both scales against game ground truth, namely the objective state recoverable from the pair's own prompt — i.e. the dialogue history up to the correction point — such as the letter feedback revealed so far in wordle or the exits listed for the current room in textmapworld. Each correction receives a verdict as either valid, invalid or unverifiable, where the latter marks cases

[4] Given %Played-Quality Score pairs, for the games in the reference games group — namely imagegame, matchit and referencegame — the untuned model achieved 75.0-88.33, 100-100, 100-80 respectively, while the SFT one scored 100-88.5, 100-100, 80-100.

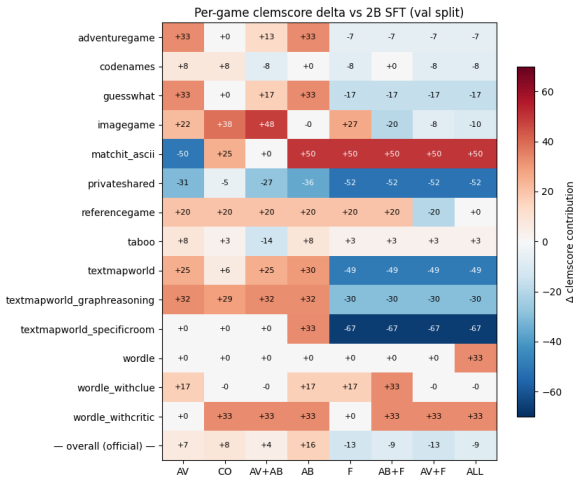

Figure 3: Per-game clemscore Δs w.r.t. the 2B SFT baseline on the playpen validation split. Rows represent games, while columns are training conditions (CO = chosen-only control), and the bottom row reports the official overall clemscore Δs. Redder cells indicate gains over the baseline, bluer ones losses.

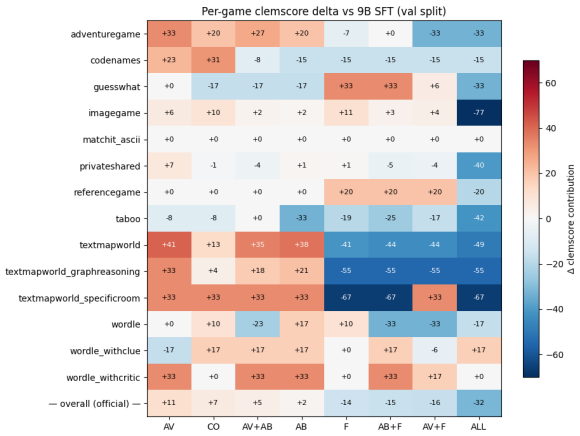

Figure 4: Per-game clemscore Δs w.r.t. the 9B SFT baseline on the playpen validation split. Rows represent games, while columns are training conditions (CO = chosen-only control), and the bottom row reports the official overall clemscore Δs. Redder cells indicate gains over the baseline, bluer ones losses.

in which no move was proposed at all, hence there is no material to score.

Note that this evaluation is based on objective, machine-readable ground truth, so it allows to test legality, rule compliance, feedback consistency and overall soundness of moves, but not strategic optimality. For this same reason, the depth of each check is dictated by what the game makes mechanically decidable. Specifically, the wordle variants, codenames and textmapworld allowed for a more precise evaluation, since they expose formal ground truth through the feedback players receive from the programmatic game master at each turn, while for all the other game families corrections evaluation could rely solely on adherence to the required answer format. The checks conducted by the per-game validator thus were:

- **Feedback consistency (wordle family):** the suggested guess must be a well-formed five-letter word that keeps all letters confirmed green, avoids letters known absent, and does not replay a yellow letter in the same position, given the feedback revealed before the correction point.
- **Board rules (codenames):** for clue-giver corrections, the suggested clue must be a single word absent from the board and the stated targets must belong to the team's words, while for guesser corrections, each suggested word must appear in the candidate list, respect the stated guess limit, and differ from the clue word.
- **Move legality (textmapworld family):** the suggested direction must be among the exits listed for the current room.
- **Format conformance (remaining games):** the correction must match the game's required answer format, specified in its rules.

In addition to our strictly programmatic validation, we note that GPT-5.2 achieved a clemscore of 84.19 at high effort, indicating how the model is an overall capable player beyond the dimensions on which its corrections were evaluated [5].

The results of our validation, outlined in Table 4, qualify the corrections as reliable, as both kinds of pairs receive near-ceiling scores across sizes, with the lowest percentage of valid corrections —

[5] Refer to clembench's official website for the model's full results on the benchmark

Pair set	n	valid	invalid	unverif.
9B failed	286	99.0%	0.7%	0.3%
9B aborted	42	85.7%	14.3%	0.0%
2B failed	325	98.8%	1.2%	0.0%
2B aborted	79	92.4%	7.6%	0.0%

Table 4: Validation scores percentages for the GPT-generated corrections, by pair set and scale. A single correction — a wordle one containing no suggested move — could not be decided against ground truth and was thus scored as unverifiable.

namely, 85.7% — belonging to the AB pairs for the 9B model.

We signal how these scores argue against correction quality as the source of the F sets' damage, as the corrections for failed rounds achieved the highest percentages of positive scores for both the 9B and 2B training sets (99.0% and 98.8% of valid corrections, respectively). In the textmapworld family, the largest source of per-game losses in the F pairs, every 9B suggestion is legal (86 of 86), and 85 out of 86 are valid at 2B. Moreover, validity and damage to performance dissociate across pair sets, as the aborted corrections are markedly less valid (85.7% at 9B, 92.4% at 2B), yet the aborted sets did not produce the regressions that the near-perfect failed sets did.

The same dissociation holds within the failed sets across games, as per-game correction validity does not correlate with per-game score change (Spearman $\rho = -0.24$, $p = 0.43$, $n = 13$).