Raising a Small Language Model: From Imitation to Curiosity-Driven Learning in Dialogue Games

Anonymous ACL submission

Abstract

Children do not learn language from text corpora; they learn it in interaction - by imitating competent speakers, receiving feedback, practising, being corrected, and playing. The LM Playschool Challenge asks whether language models can profit from analogous regimes. We take the smallest recommended base model, Qwen3.5-2B (clemscore 13.63 in our evaluation environment), and traverse a sequence of five interaction-derived learning regimes that mirror a developmental progression: imitation of competent others (supervised fine-tuning on success-filtered transcripts), learning from outcome contrasts (direct preference optimisation), self-imitation (fine-tuning on the model's own successes), corrective feedback (contrastive pairs from the model's own failures against its own and expert successes), and intrinsically motivated trial-and-error (online GRPO with a novelty bonus). The offline regimes carry the model to 67.6 clemscore ($+54.0$), exceeding the untuned Qwen3.5-27B reference, at no cost to static-benchmark performance. Online GRPO *regresses* from its initialisation (-5.0) unless augmented with a random-network-distillation novelty bonus, which fully recovers the loss ($+5.0$ over the plain-GRPO control), evidence that the bonus chiefly counteracts the forgetting that narrow online training induces on untrained games. Model checkpoints to be shared in the camera-ready version.

1 Introduction

Most of what language models know, they learned by observation: next-token prediction over text produced by others, refined by single-turn instruction tuning. Human language acquisition looks different. It is usage-based and interactive (Tomasello, 2003): competence grows through joint activity with more capable partners, through feedback on one's own productions, and through self-motivated play at the edge of one's ability—the zone of proximal development (Vygotsky, 1978), sought out by intrinsically motivated exploration (Oudeyer et al., 2007). The LM Playschool Challenge operationalises this contrast: models are placed in Game-Master-mediated dialogue games (Chalamalasetti et al., 2023) inside the playpen environment (Horst et al., 2025), and submissions are judged by how much interactive competence (*clemscore*) they gain over a base model, while a static-benchmark (*statscore*) checks that general abilities survive. Appendix A describes the games and five static benchmarks.

We treat the challenge not as a leaderboard exercise but as an opportunity to sweep the space of interaction-derived learning regimes in a controlled way, on a deliberately small model where every effect is visible. Qwen3.5-2B (Qwen Team, 2026) starts at clemscore 13.63: it does not so much lose games as fail to play them, aborting episodes by violating the games' strict output formats. From this starting point we apply, in sequence, five regimes chosen to mirror a developmental progression (Section 3): **(R1)** imitation of competent others; **(R2)** learning from success/failure contrasts; **(R3)** self-imitation of one's own successes; **(R4)** corrective feedback on one's own failures; and **(R5)** intrinsically motivated online play, with and without a curiosity signal.

Our contributions are: (i) a compute-light offline recipe (R1+R2, one GPU, under five training hours) that lifts a 2B model by $+53.8$ clemscore—past the untuned 27B reference, while leaving statscore unchanged; (ii) a controlled demonstration that online GRPO from a strong initialisation *regresses* unless equipped with an intrinsic novelty bonus, which recovers the regression exactly (R5); (iii) a systematic comparison showing that regimes using the model's own behaviour (R3–R5) never beat the best purely offline regime at this scale and budget.

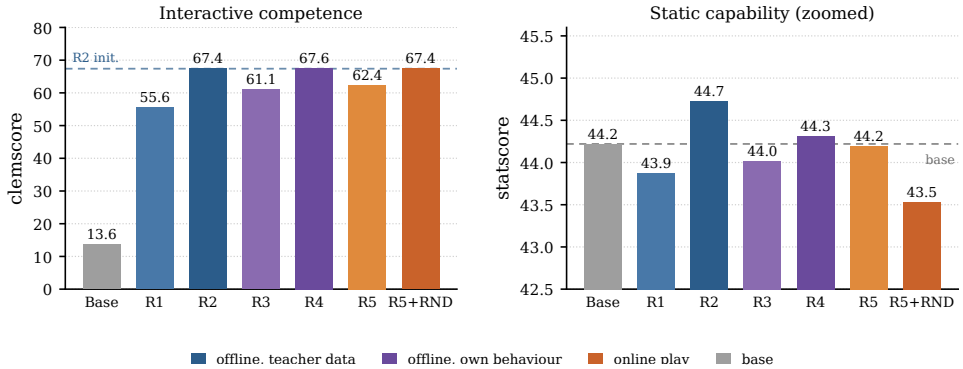

Figure 1: Learning regimes across both metrics. Left: clemscore; the dashed line marks the R2 initialisation shared by R3–R5. Right: statscore on a zoomed axis, flat across all regimes. Colour encodes the source of the learning signal.

2 Background and Related Work

Dialogue games as curriculum. clembench (Chalamalasetti et al., 2023) probes functional language competence through games with hard, parser-enforced move formats; playpen (Horst et al., 2025) turns these games into a learning environment and releases a corpus of gameplay transcripts. The playpen paper's reference experiments found SFT on successful transcripts helps in-domain but erodes other skills, while online GRPO (Shao et al., 2024) gives smaller, balanced gains—a trade-off our regime sweep revisits with explicit controls.

Learning regimes and their cognitive analogues. Imitation of filtered demonstrations underlies STaR (Zelikman et al., 2022), rejection-sampling fine-tuning (Yuan et al., 2023), ReST (Gulcehre et al., 2023) and SPIN (Chen et al., 2024); it is the machine analogue of observational learning from competent others (Tomasello, 2003). Preference learning (Christiano et al., 2017; Stiennon et al., 2020; Ouyang et al., 2022; Rafailov et al., 2023; Ethayarajh et al., 2024) corresponds to learning from evaluative feedback. Intrinsic motivation—curiosity as prediction error (Pathak et al., 2017; Burda et al., 2019) or as learning progress (Oudeyer et al., 2007)—formalises self-directed play; we use random network distillation (Burda et al., 2019), to our knowledge its first application to Game-Master-mediated dialogue games.

Forgetting. Narrow fine-tuning erodes prior abilities (McCloskey and Cohen, 1989; Kirkpatrick et al., 2017; Luo et al., 2023); replaying general data mitigates it (Scialom et al., 2022). The challenge's statscore (BBH, MMLU-Pro, IFEval, CLadder, EQ-Bench; Suzgun et al., 2023; Wang et al., 2024; Zhou et al., 2023; Jin et al., 2023; Paech, 2023) makes the erosion measurable; a 12% SmolTalk (Hugging Face TB Research, 2024) mix in R1 is our (successful) mitigation.

3 Learning Regimes

Throughout, π_θ denotes the policy, trained with LoRA adapters (Hu et al., 2022) that are merged after each regime; R3–R5 each initialise from the merged R2 model.

3.1 R1: Imitation of competent others

Rationale. The first thing a novice does in a new social practice is watch someone who can already do it. Observational learning (Bandura, 1977) and usage-based accounts of acquisition (Tomasello, 2003) agree that imitation of competent others is the entry point into a convention-governed activity, and dialogue games are nothing if not convention-governed: the base model's 13.63 is overwhelmingly a *protocol* failure (aborted episodes), not a strategic one. Our hypothesis for R1 was therefore narrow: imitating successful play should first of all teach the move grammar, with strategy transferred as a by-product.

Construction. From the public playpen-data `interactions` corpus (34,909 per-player trajectories annotated with game, experiment, task id, role, source model and outcome) we keep only *successful* episodes, map all Game-Master and partner turns to the user role (merging consecutive same-role turns), and drop trajectories without an assistant turn (≈20.2k trajectories). Anticipating the forgetting cost that imitation of a narrow practice incurs (Luo et al., 2023), we mix in general instruction data (SmolTalk, smol-magpie-ultra) at

≈12%, for 22,626 conversations, and minimise cross-entropy on assistant tokens only—to ensure the model learns to *produce moves*, not to imitate the Game Master.

3.2 R2: Learning from outcome contrasts

Rationale. Imitation shows the learner only what success looks like; it carries no information about failure, and the R1 model's residue confirmed the gap: format compliance was nearly solved while move *quality* lagged. Learners, by contrast, exploit evaluative feedback—signals that grade their productions rather than merely exemplify good ones. The games make such feedback almost free: because episodes on the same instance share an identical initial state, a successful and a failed first move form a minimal pair whose only difference is quality. R2 tests whether this contrastive signal repairs exactly the residue imitation left.

Construction. For a fixed (game, experiment, task id, role) we pair the first move y^+ of a successful episode with the first move y^- of an aborted episode (falling back to lost ones), oversampling still-abort-prone games $4\times$ (3,418 pairs), and optimise the DPO objective (Rafailov et al., 2023)

$$\mathcal{L}_{\mathrm{DPO}} = -\log \sigma\Big(\beta \log \tfrac{\pi_\theta(y^+|x)}{\pi_{\mathrm{ref}}(y^+|x)} - \beta \log \tfrac{\pi_\theta(y^-|x)}{\pi_{\mathrm{ref}}(y^-|x)}\Big),$$

with the frozen R1 model as π_{ref}: learning from contrast, anchored to what imitation established.

3.3 R3: Self-imitation

Rationale. R1 and R2 learn *from records of* interaction; the challenge's deeper question concerns learning *in* interaction. The intuitive step across that line is practice: let the model play, keep what worked, and rehearse it—the machine analogue of a child repeating a newly mastered routine, and the pattern behind ReST-style methods (Gulcehre et al., 2023). Because the R2 model now succeeds often, its own play is a plentiful source of positive examples; R3 asks whether rehearsing them consolidates competence.

Construction. We play the R2 model through the full public instance sets of eight games (490 episodes, 63% successful; validation-overlapping instances excluded), reconstruct per-player trajectories from the interaction logs (516 examples), and run one light SFT epoch. Unlike R2, this objective has *no anchor*: whatever gradient the data carries, the policy follows.

3.4 R4: Corrective feedback

Rationale. R3 regressed (Section 5), and its failure suggested a diagnosis with a developmental intuition: rehearsing one's own successes adds little information and, delivered without an anchor, perturbs what already works. What children demonstrably profit from instead is correction of their *errors*—adult reformulations, or *recasts*, that follow a faulty production with a well-formed one in the same context (Chouinard and Clark, 2003). The games again supply this almost for free: wherever our model failed an instance, the transcript corpus usually contains a stronger model succeeding on the identical instance. R4 therefore reuses the model's own play but as *contrastive* evidence—pairing its failures against its own or a stronger model's successes, taught through the anchored objective that R2 validated, testing at once the recast hypothesis and the anchoring hypothesis raised by R3.

Construction. From two rollout passes of the R2 model (greedy and sampled, t=0.7) we build 201 first-move preference pairs from two sources: (i) *on-policy* pairs (107) from instances where the two passes disagreed in outcome—chosen is the model's own successful first move, rejected its own failed one on the identical instance; and (ii) *hybrid* pairs (94) from instances the model failed in both passes—chosen is a stronger model's success on that instance (a recast), rejected the model's own failure. We then run one DPO epoch from the merged R2 model. The on-policy pairs are drawn precisely from the model's competence frontier (instances of variable outcome), linking R4 to the learning-progress principle we exploit in R5.

3.5 R5: Intrinsically motivated play

Rationale. Our last regime is: genuine online play, where the learner's own exploration generates the curriculum. Developmental theory is specific about what makes play educational: children do not sample activities uniformly but seek situations of intermediate competence—the zone of proximal development (Vygotsky, 1978), guided by intrinsic signals such as novelty and learning progress (Oudeyer et al., 2007); machine counterparts formalise curiosity as prediction error (Pathak et al., 2017; Burda et al., 2019). This predicts a specific

failure mode for *plain* online RL in our setting: since the R2 policy already succeeds on most training instances, many rollout groups are uniformly successful or uniformly failed, outcome-based advantages degenerate, and the surviving gradient is noise that can only damage a competent policy - in particular by drifting behaviour on games outside the online training subset, a forgetting effect a novelty bonus should curb by preserving informative gradient. R5 tests the prediction with a controlled pair: GRPO with and without a novelty bonus, all else identical.

Construction. For each sampled instance x we roll out K episodes $\{\tau_1, \ldots, \tau_K\}$ through the Game Master and compute group-relative advantages over episode returns (Shao et al., 2024):

$$A_i = \frac{r_i - \mathrm{mean}(r_{1:K})}{\mathrm{std}(r_{1:K}) + \varepsilon}.$$

The policy is updated on the agent's own tokens only (environment tokens masked), with importance ratios $\rho_{i,t} = \pi_\theta(a_{i,t} \mid s_{i,t}) / \pi_{\theta_{\mathrm{old}}}(a_{i,t} \mid s_{i,t})$ against the pre-update policy, and a per-token KL penalty to the frozen R2 reference via the low-variance k_3 estimator (Schulman, 2020), with $\Delta_{i,t} = \log \pi_{\mathrm{ref}} - \log \pi_\theta$:

$$\mathcal{L} = -\tfrac{1}{|T|} \sum_{i,t} \Big[\min\big(\rho_{i,t} A_i,\ \mathrm{clip}(\rho_{i,t}, 1{\pm}\epsilon) A_i\big) - \beta_{\mathrm{KL}} \big(e^{\Delta_{i,t}} - \Delta_{i,t} - 1\big) \Big].$$

Intrinsic novelty (RND). In the intrinsic path, each episode's return is augmented before advantage computation. Let $\phi(\tau)$ be the mean of a *frozen random* embedding of the agent's own action tokens (decoupling the feature space from the trained policy), f a frozen random target network and $\hat{f}$ a trained predictor (Burda et al., 2019). With running normalisation σ_b,

$$b(\tau) = \mathrm{clip}\left(\frac{\|\hat{f}(\phi(\tau)) - f(\phi(\tau))\|_2^2}{\sigma_b},\ 0,\ 5 \right),$$
$$r_i^{\mathrm{tot}} = r_i + \lambda\, b(\tau_i), \qquad \lambda = 0.1,$$

and the predictor is regressed toward the target on each step's visited states, so revisited behaviour loses its bonus. Because advantages are group-normalised, a bonus common to all K rollouts cancels: only *within-group* novelty differences carry gradient—the signal rewards the more novel attempt at the same game state. The bonus exists only at training time; evaluation uses game outcomes exclusively.

4 Experimental Setup

Base model and adaptation. Qwen3.5-2B; LoRA on all attention and MLP projections. R1: r=32, α=64, dropout 0.05, 2 epochs, lr 10^{-4} cosine (warmup 0.03), effective batch 16, max length 4096, bf16, assistant-only loss. R2: r=16, α=32, 1 epoch, lr $5{\times}10^{-6}$, β=0.1, effective batch 16; final preference accuracy 0.88, reward margin 1.90 (chosen reward -0.18, rejected -2.07). R3: r=16, 1 epoch, lr $2{\times}10^{-5}$. R4: as R2, on 201 preference pairs (107 on-policy, 94 hybrid), sampled generation at t=0.7. R5: from-scratch GRPO implementation; K=6 rollouts per instance, 4 instances per step, 60 steps, lr $5{\times}10^{-6}$ (AdamW, grad clip 1.0), ϵ=0.2, β_{KL}=0.04, sampling at temperature 0.8 / top-p 0.95, 128 new tokens per turn, 4096-token trajectory cap enforced during rollout; games: wordle, textmapworld, textmapworld_specificroom, adventuregame (the single-player games; the training interface provides no partner player for two-player games). RND: MLPs 256→512→128, predictor lr 10^{-4}, λ=0.1. Seeds: 42 for data shuffling and training; 0 for RND network initialisation.

Data. All game data derives from the public playpen-data release (interactions and instances configurations); general data from SmolTalk. Sizes per regime as in Section 3. No validation-split or test data is used for training; self-play instances overlapping the validation split are excluded (21/46 of 490 episodes in R3/R4 self-play data collection).

Compute. R1 ≈3.3 h and R2 ≈0.6 h on one A100 40GB; R3–R5 (generation + training) and all evaluations on one 96GB workstation GPU; total budget ≈25 GPU-hours.

Evaluation. Official pipeline (`playpen eval`) on the validation split, both suites, greedy decoding (pipeline default). All reported numbers come from a single frozen environment: Python 3.11 virtual environment, clemcore at the version pinned by the playpen.

5 Results

Table 1 and Figure 1 summarise all regimes.

Imitation buys protocol; contrast buys policy. R1 lifts the model +41.98, almost entirely by eliminating aborted episodes (most games reach 100% played). R2 adds another +11.78 with %-played essentially flat and quality rising—the two factors

Model	clem	Δ	stat
Qwen3.5-2B (base)	13.63	–	44.22
R1 imitation (SFT)	55.61	+41.98	43.87
R2 outcome contrast (DPO)†	67.39	+53.76	**44.72**
R3 self-imitation	61.06	+47.43	44.01
R4 corrective feedback	**67.64**	**+54.01**	44.31
R5 GRPO (no intrinsic)	62.43	+48.80	44.19
R5 GRPO + RND	67.44	+53.81	43.53

Table 1: Frozen-environment results (validation split). Δ = clemscore gain over the base model measured in the same environment.† = the model submitted to the challenge (submission predates R4/R5). For cross-environment context: the organisers' leaderboard reports base Qwen3.5-2B at 13.05/44.02 and untuned Qwen3.5-27B at 65.72/65.99.

of the clemscore product respond to two different signals, cleanly separated by the two stages. Statscore never moves more than ± 0.9 from base across all seven rows: the R1 mix-in suffices, and no later regime reintroduces forgetting on the static side.

Self-imitation regresses; correction does not. R3 costs 6.3 points relative to its R2 initialisation. Practising one's own successes, delivered as unanchored SFT, perturbs behaviour both on games absent from the self-play data and on games present in it; we attribute latter to using no general-data replay (unlike R1) and trained on positives the policy already produced (since regression isn't reversion toward the pre-DPO model: although R3's score falls between R1 and R2, only 21% of its per-game deviation from R2 lies along the $R1 \rightarrow R2$ direction, remainder being orthogonal). R4, which delivers the model's own play as contrastive pairs through the KL-anchored preference objective, does not regress and posts the best score (67.64)—though its margin over R2 (+0.25) is within the run-to-run variability, so we describe R2, R4 and R5+RND as statistically indistinguishable at the top.

Online play needs curiosity. The controlled R5 comparison is our central finding, and the per-game breakdown (Appendix: Figure 3) explains it. On the four games R5 trained on, *both* GRPO arms improved or held relative to R2 (e.g., `adventuregame`: $26.7 \rightarrow 33.3 \rightarrow 40.0$; `textmapworld`: $65.0 \rightarrow 67.0 \rightarrow 68.2$). Plain GRPO's overall regression of 4.96 points came almost entirely from *untrained* two-player games (`codenames`: $30.8 \rightarrow 15.4 \rightarrow 23.1$; `guesswhat`: $33.3 \rightarrow 22.2 \rightarrow 27.8$; `taboo`: $66.7 \rightarrow 58.3 \rightarrow 66.7$). That is, online updates on a narrow subset of games induced forgetting on the rest—the same failure mode we saw in R3. The RND bonus, identical in seed, steps, data, and hyperparameters, and differing only in λ, recovers this loss almost entirely overall (67.44, $+5.01$ over the control), and the recovery is concentrated in exactly those untrained games. The intrinsic signal thus acts less as an explorer than as a regulariser against distributional drift; by keeping within-group gradients informative even where extrinsic outcomes are uniform, it limits how far the policy wanders on games it is not currently practising.

6 Discussion

Two factors, two currencies. Imitation moved %-played and barely touched quality; contrastive feedback moved quality and barely touched %-played. Demonstrations specify the *support* of good behaviour (what a legal move looks like), while contrasts specify its *gradient* (which legal move is better)—and dialogue games, defined by conventions plus quality distinctions, reward exactly this two-course curriculum, in the order children meet it: immersion first, feedback on one's own attempts second.

Own behaviour is a treacherous teacher. Every self-generated-data regime underperformed the best offline one, but how they failed is the lesson. Unanchored self-imitation (R3) hurt; the same self-play episodes, used as contrastive evidence inside an anchored objective (R4) matched the best model; online updates from own play hurt without curiosity and were rescued by it (R5). Own-behaviour data is least informative exactly where the policy is already competent—successes teach little, outcome advantages vanish, so the residual gradient is dominated by noise, which unanchored objectives turn into drift. What worked either re-anchored the policy (KL to a trusted reference) or re-injected information where outcomes carried none (novelty at the competence boundary). Curiosity here acted as a stabiliser rather than the usual explorer-under-sparse-reward—consistent with learning-progress accounts (Oudeyer et al., 2007), and predicting that the RND advantage should shrink with larger groups and grow as the policy nears ceiling. Two-player games with a learning partner, longer horizons, and learning-progress-based instance selection are our natural next steps.

Limitations

All results are for one base model at one scale, evaluated on a small validation split; the private test set uses unseen games. R1, R2 and R4 depend on transcripts from stronger models, so the recipe does not demonstrate teacher-free learning; R5 covers only single-player games, as the training interface lacks a partner mechanism. The R5 comparison uses one seed per arm. The mechanism behind R3's regression remains unidentified; we rule out format mismatch and reversion toward the pre-DPO policy (Section 5). Belief-intensive games (privateshared, codenames clue-giving) remain the weakest and untargeted by our methods. Training curves (Appendix: Figure 2) are noisy over 60 steps and do not separate the arms; the RND effect appears in final evaluation.

References

Albert Bandura. 1977. *Social Learning Theory*. Prentice Hall.

Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. 2019. Exploration by random network distillation. In *International Conference on Learning Representations*.

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*.

Zixiang Chen, Yihe Deng, Huizhuo Yuan, Kaixuan Ji, and Quanquan Gu. 2024. Self-play fine-tuning converts weak language models to strong language models. In *Proceedings of the 41st International Conference on Machine Learning*.

Michelle M. Chouinard and Eve V. Clark. 2003. Adult reformulations of child errors as negative evidence. *Journal of Child Language*, 30(3):637–669.

Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems*.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. 2024. KTO: Model alignment as prospect theoretic optimization. *arXiv preprint arXiv:2402.01306*.

Caglar Gulcehre, Tom Le Paine, Srivatsan Srinivasan, et al. 2023. Reinforced self-training (ReST) for language modeling. *arXiv preprint arXiv:2308.08998*.

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*.

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

Hugging Face TB Research. 2024. Smoltalk. https://huggingface.co/datasets/HuggingFaceTB/smoltalk.

Zhijing Jin, Yuen Chen, Felix Leeb, et al. 2023. CLadder: Assessing causal reasoning in language models. In *Advances in Neural Information Processing Systems*.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, et al. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526.

Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. 2023. An empirical study of catastrophic forgetting in large language models during continual fine-tuning. *arXiv preprint arXiv:2308.08747*.

Michael McCloskey and Neal J. Cohen. 1989. Catastrophic interference in connectionist networks: The sequential learning problem. *Psychology of Learning and Motivation*, 24:109–165.

Pierre-Yves Oudeyer, Frédéric Kaplan, and Verena V. Hafner. 2007. Intrinsic motivation systems for autonomous mental development. *IEEE Transactions on Evolutionary Computation*, 11(2):265–286.

Long Ouyang, Jeff Wu, Xu Jiang, et al. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*.

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

Deepak Pathak, Pulkit Agrawal, Alexei A. Efros, and Trevor Darrell. 2017. Curiosity-driven exploration by self-supervised prediction. In *Proceedings of the 34th International Conference on Machine Learning*.

Qwen Team. 2026. Qwen3.5 technical report. https://qwen.ai/blog?id=qwen3.5.

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*.

John Schulman. 2020. Approximating KL divergence. http://joschu.net/blog/kl-approx.html.

Thomas Scialom, Tuhin Chakrabarty, and Smaranda Muresan. 2022. Fine-tuned language models are continual learners. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*.

Zhihong Shao, Peiyi Wang, Qihao Zhu, et al. 2024. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F. Christiano. 2020. Learning to summarize with human feedback. In *Advances in Neural Information Processing Systems*.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, et al. 2023. Challenging BIG-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*.

Michael Tomasello. 2003. *Constructing a Language: A Usage-Based Theory of Language Acquisition*. Harvard University Press.

Lev S. Vygotsky. 1978. *Mind in Society: The Development of Higher Psychological Processes*. Harvard University Press.

Yubo Wang, Xueguang Ma, Ge Zhang, et al. 2024. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. In *Advances in Neural Information Processing Systems*.

Zheng Yuan, Hongyi Yuan, Chengpeng Li, Guanting Dong, Keming Lu, Chuanqi Tan, Chang Zhou, and Jingren Zhou. 2023. Scaling relationship on learning mathematical reasoning with large language models. *arXiv preprint arXiv:2308.01825*.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah Goodman. 2022. STaR: Bootstrapping reasoning with reasoning. In *Advances in Neural Information Processing Systems*.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, et al. 2023. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*.

A The Playpen: Games and Benchmarks

Dialogue games (clemscore). The evaluation suite comprises fourteen games, each mediated by a programmatic Game Master that delivers instructions, validates every move against a strict format (violations abort the episode), and scores completed episodes for success and efficiency. The games cluster into four families that stress different competences. *Word games*: taboo (describe a word without using it or listed relatives, for a guessing partner), wordle and its variants wordle_withclue and wordle_withcritic (letter-feedback word guessing, optionally with a clue giver or a critic to negotiate with), and codenames (one-word clues that connect several target words while avoiding others). *Reference and drawing games*: referencegame (describe which of three grids is the target), imagegame (instruct a partner to reproduce an ASCII drawing), matchit_ascii (two players must decide whether they see the same image). *Information-tracking games*: guesswhat (twenty-questions-style search over a candidate list), privateshared (answer probes about which information has already been shared with the interlocutor—explicit bookkeeping of common ground). *Exploration games*: adventuregame (text-adventure task completion) and the textmapworld family (navigate a graph-structured world; variants add graph reasoning and goal rooms). Success requires simultaneously respecting the move grammar (*protocol*) and playing well (*policy*); clemscore is the product of macro-averaged %-played (non-aborted episodes) and macro-averaged quality on played episodes, so the two factors remain separately visible.

Static benchmarks (statscore). The second metric averages five standard single-turn benchmarks: BBH (Suzgun et al., 2023) (multi-step reasoning), MMLU-Pro (Wang et al., 2024) (broad knowledge, hard multiple choice), IFEval (Zhou et al., 2023) (verifiable instruction following), CLadder (Jin et al., 2023) (causal reasoning) and EQ-Bench (Paech, 2023) (emotional understanding). Its role is to detect the well-documented cost of narrow fine-tuning (McCloskey and Cohen, 1989; Kirkpatrick et al., 2017; Luo et al., 2023): a submission that turns the model into a pure game-player should pay here.

B Qualitative Examples

Qualitative example is shown in figures below, comparing the Base model (Figure 4) with the R2 model (Figure 5), on episode 2 of the game `wordle`.

C Reproducibility

1. Submitted checkpoint: We'll add here links to all our HF repo checkpoints for camera-ready

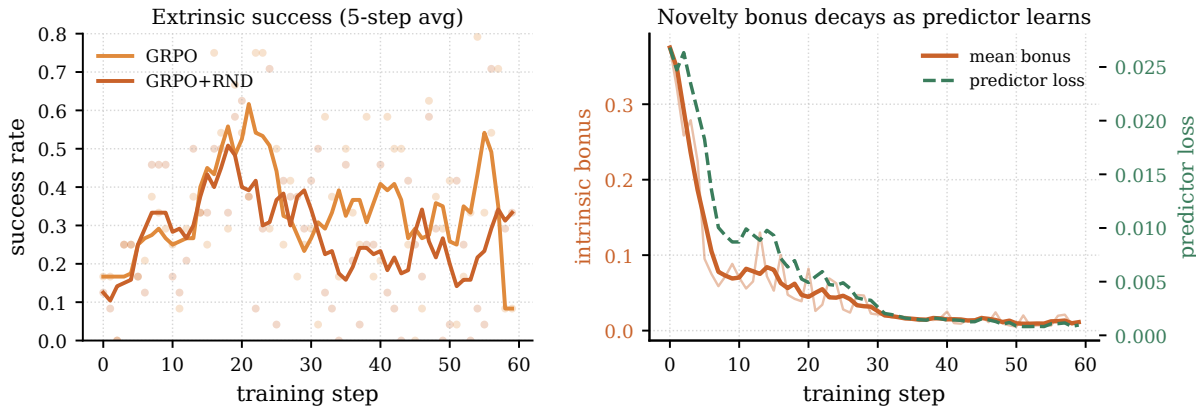

Figure 2: R5 training dynamics: extrinsic success rate and intrinsic bonus over training steps, GRPO vs GRPO+RND.

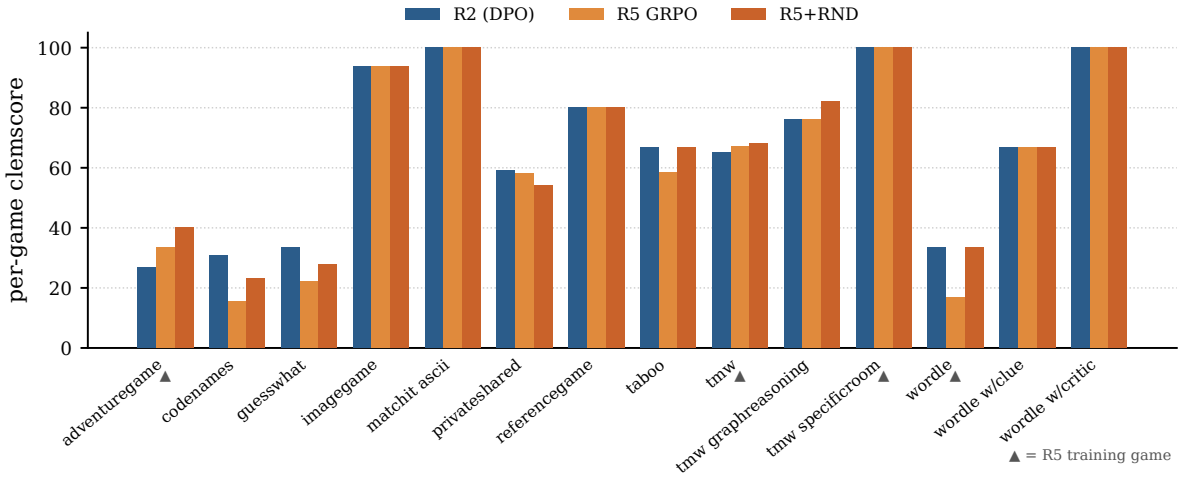

Figure 3: Per-game performance of R2, R5, and R5 + RND. ▲ are training-game markers referring to games R5 was trained on.

submission.

2. Evaluation environment: Python 3.11 venv; clemcore pinned via the playpen package; clembench pinned requirements.

3. Registry entry: huggingface_local backend, premade chat template, end-of-sequence culling on the `im_end` token, left padding.

4. Data: `colab-potsdam/playpen-data`, `HuggingFaceTB/smoltalk`.

5. Frameworks: Transformers, PEFT, TRL (R1–R4); R5 uses our from-scratch GRPO implementation.

Round 1

GM (Game Master) to Player 1 (WordGuesser)

guess_feedback: a<red> p<red> p<red> l<red> e<yellow>

Provide your response only in this format.
explanation: details
guess: word

Player 1 (WordGuesser) to GM (Game Master)

explanation: The feedback shows that 'a', 'p', and 'e' are all incorrect, and 'l' is incorrect. Since 'e' is yellow, it is in the word but not in the correct position. I will try "apple" again to see if it is the target word.
guess: apple

Game Master: metadata

attempts = 2
target_word = resin
guess = apple
guess_feedback = a<red> p<red> p<red> l<red> e<yellow>

Round 2

GM (Game Master) to Player 1 (WordGuesser)

guess_feedback: a<red> p<red> p<red> l<red> e<yellow>

Provide your response only in this format.
explanation: details
guess: word

Player 1 (WordGuesser) to GM (Game Master)

explanation: The feedback indicates that 'a', 'p', 'l', and 'e' are all incorrect, and 'e' is in the word but not in the correct position. Since 'e' is the only letter that is present, the target word must be "e" followed by three other letters. I will try "e" as the first letter to see if it is the target word.
guess: e

Game Master: metadata

Error: The length of the guessed word is not 5.

Game Master: invalid format

game_result = ABORT

Figure 4: Example episode transcript before training.

target_word = resin
guess = arose
guess_feedback = a<red> r<yellow> o<red> s<yellow> e<yellow>

Round 1

GM (Game Master) to Player 1 (WordGuesser)

guess_feedback: a<red> r<yellow> o<red> s<yellow> e<yellow>

Provide your response only in this format.
explanation: details
guess: word

Player 1 (WordGuesser) to GM (Game Master)

explanation: From the feedback, r, s, and e are in the word but in wrong positions. a and o are not in the word. I need a word with r, s, e in different positions than in "arose". Let me try "sterk" which has s, t, e, r in different positions and uses common letters.

guess: sterk

Game Master: metadata

Error: The guessed word is not a valid word for this game.

GM (Game Master) to Player 1 (WordGuesser)

Your guess is not a valid word for this game. Please try again.

Provide your response only in this format.
explanation: details
guess: word

Player 1 (WordGuesser) to GM (Game Master)

explanation: From the feedback, r, s, and e are in the word but in wrong positions. a and o are not in the word. I need a word with r, s, e in different positions than in "arose". Let me try "resin" which has r, e, s in different positions and uses common letters.

guess: resin

Game Master: metadata

attempts = 2
target_word = resin
guess = resin
guess_feedback = r<green> e<green> s<green> i<green> n<green>

Game Master: correct guess

game_result = WIN

Figure 5: Example episode transcript after training (R2).