Adapting SCoRe to Single-Player Dialogue Games: Initial Experiments on PLAYPEN

Neda Foroutan[1,2*], **Valhari Meshram**[1*]
[1]Potsdam Universität
[2]Technische Universität Berlin
{neda.foroutan, valhari.meshram}@uni-potsdam.de
[*]These authors contributed equally as first authors.

Abstract

Large Language Models (LLMs) are increasingly deployed as autonomous agents that solve tasks through multi-step interactions. However, most post-training methods are designed for single-turn reasoning or instruction-following tasks and provide limited supervision for long-horizon conversational decision making. In this work, we investigate whether Self-Correction via Reinforcement Learning (SCoRe), originally proposed for single-step mathematical reasoning and code generation, can improve agentic performance in multi-turn dialogue games. We adapt SCoRe to the PLAYPEN benchmark by fine-tuning Qwen3.5-9B on five single-player dialogue games using interaction-level rewards and iterative self-correction. Experimental results show that the resulting model substantially improves performance on the trained single-player games and further generalizes to unseen multi-player dialogue games, increasing the overall ClemScore from 41.18 to 43.31. Although a slight decrease is observed on the aggregate StatScore, the model largely retains its general language understanding and reasoning capabilities. These findings demonstrate that self-correction reinforcement learning is an effective approach for improving long-horizon conversational agent behavior beyond single-step reasoning tasks.

1 Introduction

LLMs have achieved remarkable performance across a wide range of natural language processing tasks, including question answering (Lucas et al., 2024; Borroto Santana et al., 2025), code generation(Dolcetti et al., 2026; Dong et al., 2024), and instruction following(Dong et al., 2025; Wu et al., 2024). Most existing LLMs are trained to generate a single response given an input prompt (Marvin et al., 2024), making them highly effective for both traditional single-turn applications (Nadi et al., 2024) and instruction-following (Heo et al., 2025) tasks. More recently, LLMs have also been deployed as autonomous agents that interact with external environments by planning, reasoning, and taking actions over multiple steps to accomplish complex tasks (Kannan et al., 2024; Janakiraman, 2026).

Unlike single-turn settings, agentic applications require LLMs to maintain context, adapt to feedback, and make decisions throughout an interaction (Ouyang et al., 2026). However, most foundation models are pre-trained on static text corpora and instruction-tuned on single prompt-response pairs, providing limited supervision for multi-step conversational decision making. Reinforcement learning has therefore become a common approach for improving agentic behavior by optimizing models based on interaction outcomes rather than individual responses (Horst et al., 2025; Zhou and Zanette, 2024).

The LM Playschool Challenge[1] investigates whether training on multi-step conversational interactions, referred to as *dialogue games*, can improve the agentic capabilities of LLMs. In these games, an LLM interacts with a game environment over multiple turns, where each action influences subsequent observations and the final task outcome. This setting provides a realistic benchmark for evaluating planning, reasoning, and decision-making in interactive environments beyond conventional static benchmarks.

In this work, we, as the Dialogue Architects team, adapt Self-Correction via Reinforcement Learning (SCoRe) (Kumar et al., 2025) to multi-turn dialogue games by fine-tuning Qwen3.5-9B on five single-player games from the PLAYPEN benchmark. Unlike the original SCoRe framework, which focuses on single-step reasoning and code generation, our approach applies iterative self-correction to long-horizon conversational interac-

[1]https://lm-playschool.github.io/challenge/

tions. Experimental results show that SCoRe improves performance on both trained single-player games and unseen multi-player games, increasing the overall ClemScore (+2.13), while maintaining competitive performance on general language understanding and reasoning benchmarks. These findings demonstrate that self-correction reinforcement learning is an effective approach for improving long-horizon conversational agent behavior.

2 Related Work

The PLAYPEN framework (Horst et al., 2025) introduces a benchmark for training and evaluating LLMs as conversational agents in multi-turn dialogue games. Unlike traditional instruction-following benchmarks, PLAYPEN evaluates models on long-horizon interactions where each action influences future observations and the final task outcome. Performance is measured using *ClemScore* for interactive gameplay and *StatScore* for general language understanding and reasoning. To improve agentic performance, the PLAYPEN authors investigated Supervised Fine-Tuning (SFT), Direct Preference Optimization (DPO), and Group Relative Policy Optimization (GRPO). While SFT and DPO improved performance on games seen during training, they generalized poorly to unseen games and often reduced StatScore. GRPO achieved the best trade-off by improving interactive gameplay while largely preserving general language capabilities.

Self-correction has recently emerged as an effective paradigm for reinforcement learning across a variety of sequential decision-making tasks, including autonomous driving (Guo et al., 2026), image captioning (Zhang et al., 2025), mathematical reasoning (Xiong et al., 2025), and search agents (Li et al., 2026). These methods enable models to iteratively revise intermediate decisions instead of relying solely on final outcome rewards.

Another self-correction approach is SCoRe (Self-Correction via Reinforcement Learning) (Kumar et al., 2025), which introduces a multi-turn reinforcement learning framework that explicitly trains language models to improve their own responses through iterative self-correction. Rather than relying on supervised correction traces or external teacher models, SCoRe performs online reinforcement learning on self-generated trajectories and employs reward shaping to encourage successful self-correction while preventing behavior collapse. The method achieved state-of-the-art performance on the MATH and HumanEval benchmarks, substantially improving mathematical reasoning and code generation through a single self-correction step.

Our work extends SCoRe to a substantially different setting. While the original framework was developed for single-step reasoning and coding tasks, we apply it to multi-turn dialogue games in the PLAYPEN benchmark, where the model must maintain conversational context, adapt its decisions over multiple interaction steps, and optimize for episode-level success. Our results show that SCoRe improves interactive dialogue-game performance and is effective beyond single-step reasoning tasks and can enhance long-horizon conversational agent behavior.

3 Methodology

3.1 SCoRe Learning Algorithm

SCoRe (Kumar et al., 2025) is a two-stage reinforcement learning framework designed to improve a language model's ability to self-correct. During training, the model first generates an initial attempt (y_1) and then generates a second attempt (y_2) conditioned on the original task, the complete trajectory of the first attempt, and a self-correction prompt. The objective of SCoRe is to encourage the second attempt to correct mistakes made in the first attempt while maintaining stable optimization throughout training. Figure 1 provides an overview of the two-stage SCoRe training framework.

Stage I – Constrained Initialization using KL-Divergence: The policy model is initialized using reinforcement learning to improve the accuracy of the second attempt while keeping the first attempt close to the original base model. A KL-divergence regularization constrains the first attempt to remain similar to the base model, allowing the second attempt to improve independently. This stage reduces the tendency to couple the two attempts and provides a stable initialization for subsequent self-correction training.

Stage II – Reward-Shaped Multi-Turn RL: Starting from the Stage I policy, the model is further optimized using a reward-shaping objective that jointly encourages the correctness of the second attempt and meaningful improvement over the first attempt. A reward bonus is assigned when the second attempt successfully corrects an incorrect first attempt, while a large negative penalty is applied when a correct first attempt is changed into an

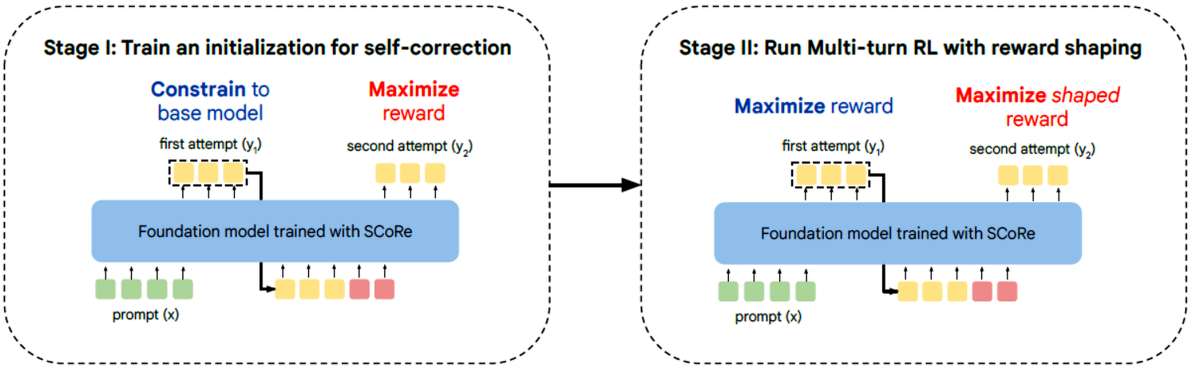

Figure 1: Overview of the two-stage SCoRe training framework: Source: (Kumar et al., 2025). Stage I trains a stable initialization by constraining the first attempt to remain close to the base model while optimizing the second attempt. Stage II jointly optimizes both attempts using reward-shaped reinforcement learning to encourage successful self-correction.

incorrect one. This reward design explicitly encourages effective self-correction while discouraging unnecessary revisions.

3.2 Implementation

We used Qwen3.5-9B[2] as the base policy model and fine-tuned it using the two-stage SCoRe framework (Kumar et al., 2025) on the single-player dialogue games from the PLAYPEN benchmark[3]. We employed 4-bit QLoRA for parameter-efficient fine-tuning and merged the learned LoRA weights into the base model after training. Unlike the original SCoRe work, which focused on single-step reasoning and code generation, we adapt SCoRe to multi-turn dialogue games, where each self-correction attempt corresponds to an entire gameplay episode rather than a single generated response. The training data consists of five single-player PLAYPEN games: *adventuregame*, *textmapworld*, *textmapworld_graphreasoning*, *textmapworld_specificroom*, and *wordle*. The total number of training instances is 161, with the breakdown by game reported in Table 4 in Appendix A.

For each training episode, the model first generates an initial attempt (y_1) by completing an entire dialogue-game episode. The complete interaction trajectory together with the episode outcome (successful, unsuccessful, or aborted) is then provided through a self-correction prompt, enabling the model to reflect on its previous decisions before generating a second attempt (y_2). Figure 2 shows the self-correction prompt template used to transition from the first attempt to the second attempt.

[2] https://huggingface.co/Qwen/Qwen3.5-9B

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

Episode-level rewards were computed using the success signals returned by each CLEMcore game evaluator. A reward of $+1$ was assigned when the policy model successfully completed the game, 0 when it failed to solve the game, and -1 when the episode was aborted due to a rule violation or invalid interaction.

To facilitate reproducibility, the training scripts are publicly available at https://github.com/NedaForoutan/SCoRe-to-Single_Player-Dialogue-Game, and the fine-tuned model is released at https://huggingface.co/Valhari14/score-qwen3.5-9b-playpen.

3.2.1 Experimental Setup

Training on the five single-player dialogue games took approximately 48 hours using three NVIDIA A30 GPUs on a single compute node. The GPUs were used to parallelize on-policy rollout generation, as SCoRe requires two complete game playthroughs per training episode, with an initial attempt followed by a self-correction attempt, resulting in approximately twice the generation cost of standard single-attempt reinforcement learning fine-tuning. To enable memory-efficient training, the model was loaded using 4-bit QLoRA through the Hugging Face Transformers, PEFT, and bitsandbytes libraries. Unless otherwise specified, we followed the training configuration of the original SCoRe implementation.

The Stage II reward-shaping coefficient was set to $\alpha = 10.0$, while the KL-divergence regularization weights were $\beta_1 = 0.01$ for Stage II and $\beta_2 = 0.1$ for Stage I. The learning rate was 1×10^{-5}. During training, the sampling tempera-

```
if y1.success :  "You SUCCEEDED in your previous attempt."
if y1.aborted :  "Your previous attempt was ABORTED due to a rule violation."
if y1.lost    :  "Your previous attempt was UNSUCCESSFUL."

"Below is a summary of what you did last time. Carefully review it, identify what
went wrong or could be improved, and play better this time. A fresh attempt at the
same task begins now."
```

Figure 2: Self-correction prompt template used during SCoRe training. After completing the first attempt, the model receives feedback indicating whether the episode was successful, unsuccessful, or aborted, together with an instruction to review its previous decisions before generating a second attempt.

Model	ClemScore	StatScore
Qwen3.5-9B	41.18	54.35
SCoRe_Qwen3.5-9B	43.31 ■	52.71 ■

Table 1: Performance comparison between the base **Qwen3.5-9B** model and the **SCoRe**-fine-tuned variant on the PLAYPEN benchmark. ClemScore evaluates performance on all interactive dialogue games, while StatScore is the average across BBH, CLadder, EQBench, IFEval, and MMLU-Pro.

ture was set to 0.8, the maximum number of generated tokens per attempt was 256, and Stage I and Stage II were trained for 3 and 5 epochs, respectively.

4 Results

Table 1 compares the overall performance of the base Qwen3.5-9B model and the SCoRe fine-tuned model, SCoRe_Qwen3.5-9B, on the PLAYPEN benchmark. SCoRe_Qwen3.5-9B improves the overall ClemScore from 41.18 to 43.31 (+2.13), indicating better performance on interactive dialogue games. The aggregate StatScore decreases only slightly from 54.35 to 52.71, suggesting that the reinforcement learning fine-tuning largely preserves the model's general language understanding and reasoning capabilities despite being optimized for interactive dialogue tasks.

Performance on Single-Player Dialogue Games: Table 2 presents the performance of the SCoRe fine-tuned model on the five single-player PLAYPEN dialogue games. Compared with the baseline model, SCoRe substantially improves the overall ClemScore from 39.38 to 47.31 (+7.93). Although the completion rate (% Played) remains unchanged across all games, the quality score (QS) consistently improves on every game except *Wordle*, where both models aborted or timed out on several instances. The largest improvement is observed on *AdventureGame*, where the quality score increases from 11.11 to 33.33, followed by *TextMapWorld_SpecificRoom* (39.66 to 56.45) and *TextMapWorld* (68.01 to 73.07). These results indicate that SCoRe primarily improves the quality of decision making within successful gameplay episodes on the single-player dialogue games used for training, leading to a substantial improvement in overall interactive performance.

Generalization to Multi-Player Dialogue Games: To evaluate whether the learned self-correction behaviour generalizes beyond the training games, we evaluated the model on unseen multi-player PLAYPEN dialogue games. As shown in Table 3, SCoRe improves the overall ClemScore from 41.18 to 43.31 (+2.13) and increases the average completion rate from 70.21% to 74.37%. The largest improvement is observed on *Wordle_withClue*, where the completion rate doubles from 33.33% to 66.67%, although the average quality score decreases because the additional successful-played episodes receive lower-quality solutions. SCoRe also improves the quality score on *Taboo* from 41.67 to 50.00, indicating that the learned self-correction strategy transfers effectively to this collaborative word-based game.

Performance decreases slightly on *Private-Shared*, where the quality score drops by 4.26 points, and the overall average quality score decreases modestly from 66.20 to 64.44. Moreover, the baseline and SCoRe models achieve identical performance on *ImageGame*, *MatchIt_ASCII*, *ReferenceGame*, and *Wordle_withCritic*. And both models consistently aborted or timed out on *Wordle*, *GuessWhat*, and *Codenames*.

These results suggest that training with SCoRe on single-player dialogue games improves the model's ability to generalize to unseen multi-player dialogue games, particularly by increasing task

	Overall	All,Average		AdventureGame		TextMapWorld		TMW-GR		TMW-SR	
Model	**ClemScore**	**%**	**QS**	**%**	**QS**	**%**	**QS**	**%**	**QS**	**%**	**QS**
Qwen3.5-9B	39.38	72	54.7	60	11.11	100	68.01	100	39.66	100	100
SCoRe-Qwen3.5-9B	**47.31**$^{+7.93}$	72	**65.71**$^{+11.01}$	60	**33.33**$^{+22.22}$	100	**73.07**$^{+5.06}$	100	**56.45**$^{+16.79}$	100	100

Table 2: Generalization performance of the SCoRe-Qwen3.5-9B (SCoRe-Qwen) model on **unseen single-player PLAYPEN dialogue games** after training on five single-player dialogue games. "%" denotes the percentage of successfully played games, and QS denotes the average quality score for completed games. TMW-GR and TMW-SR refer to *TextMapWorld_GraphReasoning* and *TextMapWorld_SpecificRoom*, respectively.

completion rates. While improvements in interaction quality are not uniform across all game types, the gains in overall ClemScore demonstrate that self-correction reinforcement learning transfers beyond the training games and improves long-horizon conversational decision making.

Overall, the experimental results demonstrate that SCoRe is an effective reinforcement learning strategy for improving LLMs in interactive dialogue-game environments. Fine-tuning on a small set of single-player dialogue games substantially improves in-domain performance while also transferring to unseen multi-player games. Importantly, these gains are achieved with only a modest reduction in StatScore, indicating that the improvements in agentic behaviour come without significant degradation of the model's general language understanding and reasoning capabilities.

5 Conclusion

This work adapts Self-Correction via Reinforcement Learning (SCoRe) to multi-turn dialogue games in the PLAYPEN benchmark. Unlike the original SCoRe framework, which was developed for single-step mathematical reasoning and code generation, our approach applies iterative self-correction to long-horizon conversational interactions. The model is trained on five single-player dialogue games, where each training episode consists of an initial gameplay attempt followed by a self-corrected replay of the same game instance. Experimental results show that SCoRe substantially improves performance on the trained single-player dialogue games and also generalizes to unseen multi-player games, demonstrating that self-correction learned from single-player interactions transfers to more complex collaborative settings. Overall, our findings demonstrate that self-correction reinforcement learning is a promising training paradigm for enhancing the agentic capabilities of large language models in long-horizon conversational environments. Beyond improving performance on individual dialogue games, the proposed adaptation shows that SCoRe can be successfully extended from single-step reasoning tasks to sequential decision-making settings involving multi-turn interactions.

6 Limitations

This work has several limitations. First, due to the lack of time for training SCoRe, our experiments are limited to the five single-player dialogue games provided in the PLAYPEN benchmark. Extending the approach to multi-player dialogue games requires substantially longer training times, and is therefore left for future work.

Second, we evaluate SCoRe using a single base model, Qwen3.5-9B. Because of the limited time and computational resources required for reinforcement learning fine-tuning, we were unable to investigate the effectiveness of SCoRe on additional LLMs. Evaluating the proposed approach across a broader range of open-source models with different architectures and scales would provide a better understanding of its generalizability.

References

Manuel Alejandro Borroto Santana, Katie Gallagher, Antonio Ielo, Irfan Kareem, Francesco Ricca, and Alessandra Russo. 2025. Question answering with llms and learning from answer sets. *Theory and Practice of Logic Programming*, page 1–25.

Greta Dolcetti, Vincenzo Arceri, Eleonora Iotti, Sergio Maffeis, Agostino Cortesi, and Enea Zaffanella. 2026. Helping llms improve code generation using feedback from testing and static analysis. *Discover Artificial Intelligence*, 6(1):314.

Guanting Dong, Keming Lu, Chengpeng Li, Tingyu Xia, Bowen Yu, Chang Zhou, and Jingren Zhou. 2025. Self-play with execution feedback: Improving instruction-following capabilities of large language models. In *International Conference on Learning Representations*, volume 2025, pages 39286–39313.

	Overall	All Avg.		PrivateShared		Taboo		Wordle+Clue	
Model	**ClemScore**	**%**	**QS**	**%**	**QS**	**%**	**QS**	**%**	**QS**
Qwen3.5-9B	41.18	70.21	**66.20**	80.00	**26.52**	100.00	41.67	33.33	**100.00**
SCoRe-Qwen3.5-9B	**43.31**$^{+2.13}$	**74.37**$^{+4.16}$	64.44$^{-1.76}$	80.00	22.26$^{-4.26}$	100.00	**50.00**$^{+8.33}$	**66.67**$^{+33.34}$	66.66$^{-33.34}$

Table 3: Generalization performance of the SCoRe-Qwen3.5-9B (SCoRe-Qwen) model on **unseen multi-player PLAYPEN dialogue games** after training on five single-player dialogue games. "%" denotes the percentage of successfully completed games, and QS denotes the average quality score. For brevity, only games exhibiting differences between the baseline Qwen3.5-9B and SCoRe-Qwen are reported.

Yihong Dong, Xue Jiang, Zhi Jin, and Ge Li. 2024. Self-collaboration code generation via chatgpt. *ACM Trans. Softw. Eng. Methodol.*, 33(7).

Yihong Guo, Dongqiangzi Ye, Sijia Chen, Anqi Liu, and Xianming Liu. 2026. Correctionplanner: Self-correction planner with reinforcement learning in autonomous driving. *Preprint*, arXiv:2603.15771.

Juyeon Heo, Christina Heinze-Deml, Oussama Elachqar, Kwan Ho Ryan Chan, Shirley Ren, Andrew Miller, Udhyakumar Nallasamy, and Jaya Narain. 2025. Do llms ``know"internally when they follow instructions? In *International Conference on Learning Representations*, volume 2025, pages 81339–81357.

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.

Anantharaman Janakiraman. 2026. From generative intelligence to agentic autonomy: Leveraging large language models for multi-agent reasoning, planning, and execution. *Journal of Integrated Science, AI and Engineering*, 2(1).

Shyam Sundar Kannan, Vishnunandan L. N. Venkatesh, and Byung-Cheol Min. 2024. Smart-llm: Smart multi-agent robot task planning using large language models. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 12140–12147.

Aviral Kumar, Vincent Zhuang, Rishabh Agarwal, Yi Su, JD Co-Reyes, Avi Singh, Kate Baumli, Shariq Iqbal, Colton Bishop, Rebecca Roelofs, Lei Zhang, Kay McKinney, Disha Shrivastava, Cosmin Paduraru, George Tucker, Doina Precup, Feryal Behbahani, and Aleksandra Faust. 2025. Training language models to self-correct via reinforcement learning. In *International Conference on Learning Representations*, volume 2025, pages 54523–54549.

Shiyu Li, Yang Tang, Yifan Wang, Peiming Li, and Xi Chen. 2026. Reseek: A self-correcting framework for search agents with instructive rewards. *Preprint*, arXiv:2510.00568.

Mary M Lucas, Justin Yang, Jon K Pomeroy, and Christopher C Yang. 2024. Reasoning with large language models for medical question answering. *Journal of the American Medical Informatics Association*, 31(9):1964–1975.

Ggaliwango Marvin, Nakayiza Hellen, Daudi Jjingo, and Joyce Nakatumba-Nabende. 2024. Prompt engineering in large language models. In *Data Intelligence and Cognitive Informatics*, pages 387–402, Singapore. Springer Nature Singapore.

Farhad Nadi, Hadi Naghavipour, Tahir Mehmood, Aliesya Binti Azman, Jeetha A/P Nagantheran, Kezia Sim Kui Ting, Nor Muhammad Ilman Bin Nor Adnan, Roshene A/P Sivarajan, Suita A/P Veerah, and Romi Fadillah Rahmat. 2024. Sentiment analysis using large language models: A case study of gpt-3.5. In *Data Science and Emerging Technologies*, pages 161–168, Singapore. Springer Nature Singapore.

Siru Ouyang, Jun Yan, I-Hung Hsu, Yanfei Chen, Ke Jiang, Zifeng Wang, Rujun Han, Long Le, Samira Daruki, Xiangru Tang, Vishy Tirumalashetty, George Lee, Mahsan Rofouei, Hangfei Lin, Jiawei Han, Chen-Yu Lee, and Tomas Pfister. 2026. Reasoningbank: Scaling agent self-evolving with reasoning memory. In *International Conference on Learning Representations*, volume 2026, pages 94327–94354.

Tianhao Wu, Janice Lan, Weizhe Yuan, Jiantao Jiao, Jason Weston, and Sainbayar Sukhbaatar. 2024. Thinking llms: General instruction following with thought generation. *Preprint*, arXiv:2410.10630.

Wei Xiong, Hanning Zhang, Chenlu Ye, Lichang Chen, Nan Jiang, and Tong Zhang. 2025. Self-rewarding correction for mathematical reasoning. *Preprint*, arXiv:2502.19613.

Lin Zhang, Xianfang Zeng, Kangcong Li, Gang Yu, and Tao Chen. 2025. Sc-captioner: Improving image captioning with self-correction by reinforcement learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 23145–23155.

Yifei Zhou and Andrea Zanette. 2024. Archer: training language model agents via hierarchical multi-turn rl. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

A Training Instances

Table 4 reports the number of training instances for each of the single-player games from the PLAYPEN benchmark. The training set comprises 161 instances in total, distributed across AdventureGame, TextMapWorld, TextMapWorld_GraphReasoning, TextMapWorld_SpecificRoom, and Wordle. These instances were used to fine-tune Qwen3.5-9B with Self-Correction via Reinforcement Learning (SCoRe) for multi-turn dialogue-game interactions.

Game	**# of Instances**
AdventureGame	35
TextMapWorld	45
TextMapWorld_GraphReasoning	27
TextMapWorld_SpecificRoom	27
Wordle	27
Total	**161**

Table 4: Number of training instances for each single-player PLAYPEN game used for fine-tuning of SCoRe_Qwen3.5-9B.

B Dialogue Game Transcripts

To qualitatively illustrate the effect of SCoRe, Figure 3 presents a representative self-correction episode from the *textmapworld* game. In the first attempt (y_1), the model terminates the game after revisiting previously explored rooms, failing to discover all reachable locations. After receiving the self-correction prompt shown in Figure 2, the model generates a second attempt (y_2), avoids the earlier navigation error, successfully explores the remaining room, and completes the task. This example demonstrates how SCoRe enables the model to reflect on its previous attempt and improve its decision-making over multiple interaction steps.

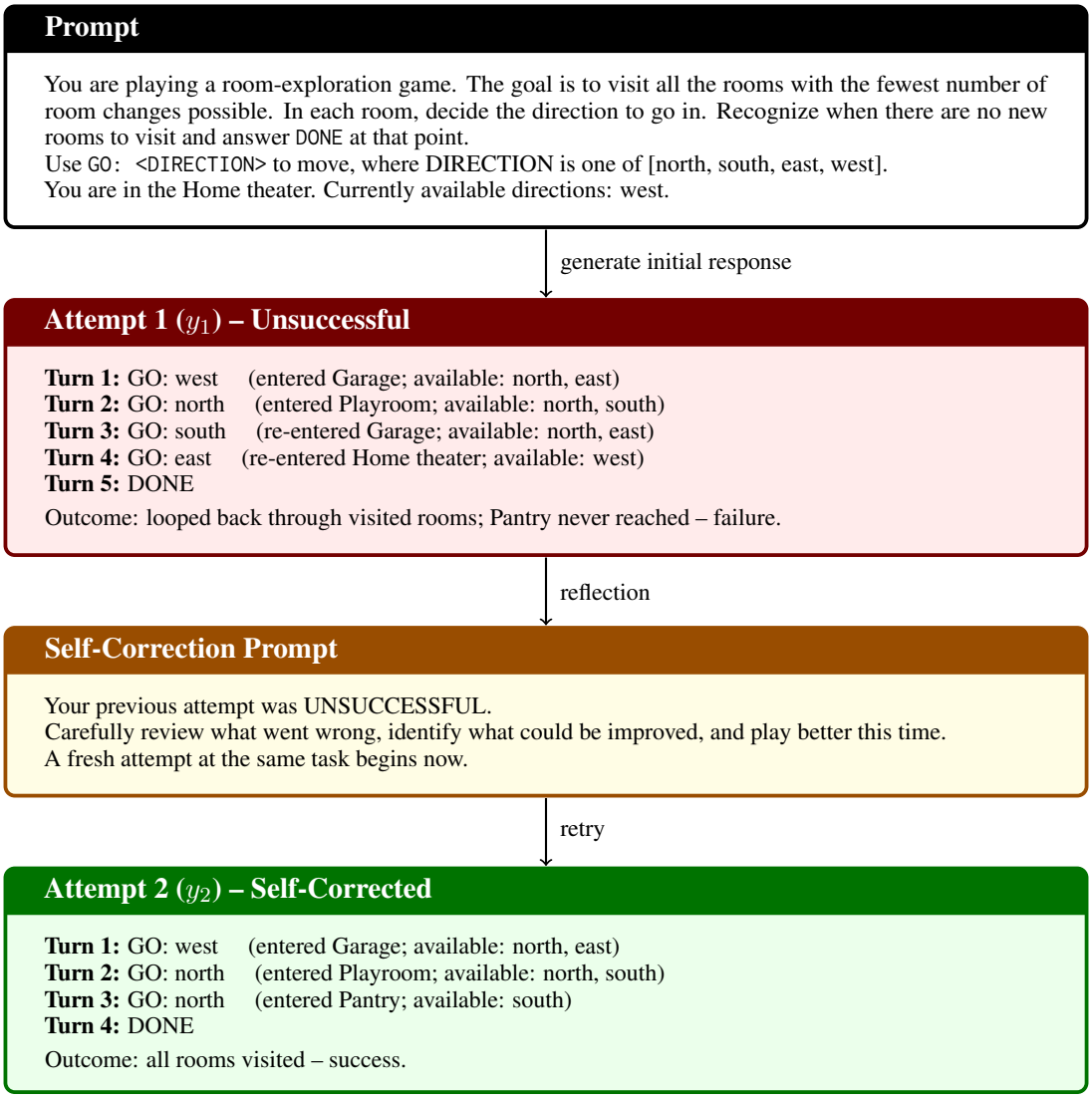

Figure 3: SCoRe self-correction example on `textmapworld`. The model first generates an unsuccessful attempt (y_1), receives the self-correction prompt, and then generates a corrected attempt (y_2) that solves the task.