

GIFT: Games as Informal Training for Generalizable LLMs

Nuoyan Lyu^{♣♡}, Bingbing Xu^{♣*}, Weihao Meng, Yige Yuan^{♣♡},
 Yang Zhang[◊], Zhiyong Huang[◊], Tat-Seng Chua[◊], Huawei Shen^{♣♡}

[♣] State Key Laboratory of AI Safety, Institute of Computing Technology, CAS

[♡] University of Chinese Academy of Sciences

[◊] National University of Singapore

{1vnuoyan23z, xubingbing, yuanyige20z}@ict.ac.cn, wmeng9@jh.edu

Abstract

While Large Language Models (LLMs) have achieved remarkable success in formal learning tasks such as mathematics and code generation, they still struggle with the "practical wisdom" and generalizable intelligence, such as strategic creativity and social reasoning, that characterize human cognition. This gap arises from a lack of informal learning, which thrives on interactive feedback rather than goal-oriented instruction. In this paper, we propose treating Games as a primary environment for LLM informal learning, leveraging their intrinsic reward signals and abstracted complexity to cultivate diverse competencies. To address the performance degradation observed in multi-task learning, we introduce a Nested Training Framework. Unlike naive task mixing optimizing an implicit "OR" objective, our framework employs sequential task composition to enforce an explicit "AND" objective, compelling the model to master multiple abilities simultaneously to achieve maximal rewards. Using GRPO-based reinforcement learning across *Matrix Games*, *TicTacToe*, and *Who's the Spy* games, we demonstrate that integrating game-based informal learning not only prevents task interference but also significantly bolsters the model's generalization across broad ability-oriented benchmarks. The framework and implementation are publicly available¹.

1 Introduction

Human intelligence arises from the interplay between formal and informal learning (Scribner and Cole, 1973). Formal learning emphasizes structured, goal-oriented education for acquiring task-specific knowledge, whereas informal learning unfolds in everyday environments through the unstructured interactions involving iterative experience and implicit feedback, enabling the acquisition of

implicit knowledge and practical wisdom that further support general and transferable intelligence (Callanan et al., 2011). In parallel, recent large language models (LLMs) have achieved remarkable success on formal learning tasks (Liu et al., 2025c; Xu et al., 2025), including mathematical reasoning (Wang et al., 2025b; Zhang et al., 2025b; Chen et al., 2025) and code generation (Seed et al., 2025; Yang et al., 2025; Wang et al., 2025a). However, the broader competencies expected of general-purpose models, including creativity, social reasoning, etc., lie beyond the scope of formal learning and call for learning mechanisms akin to informal learning.

Therefore, this prompts us to think what is the informal learning environment for LLMs. In this paper, we propose a new perspective: treating **games** as a fundamental environment for informal learning due to the following three key properties: 1) emerging from unstructured interactions involving iterative experience; 2) proceeding without explicit instructions, enabling learning without reliance on manually annotated data, overcoming the limitations of high-cost datasets; 3) serving as highly abstracted sandboxes of complex real-world interactions (Edwards et al., 2019; Roungas et al., 2019; Kriz et al., 2022), unifying a diverse set of tasks and closely aligning with the goals of informal learning (Innes and Booher, 1999; Dutta, 1999; Aumann and Hart, 1992).

Building on this insight, we introduce a new training strategy of LLM that integrates formal learning with game-based informal learning. Specifically, we train LLMs using GRPO-based reinforcement learning (Shao et al., 2024) across mathematical tasks (formal learning) and multiple game environments (informal learning). We design three representative categories of games to cover diverse capacities: single-turn games (*Matrix Games*), multi-turn two-player games (*TicTacToe*) and multi-turn multi-agent social games (*Who's the Spy*), collectively covering a broad spectrum of cog-

*Corresponding author.

¹<https://github.com/XXX/XXXX>

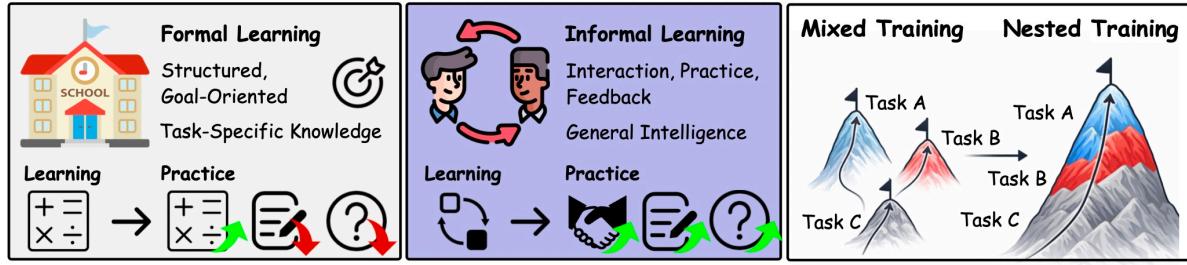


Figure 1: Overview of formal and informal learning paradigms, and a comparison between naive mixed training and the proposed nested training framework.

nitive abilities, including abstract reasoning (Lucas, 1981; Aumann and Hart, 1992), long-horizon planning (Mishra et al., 2025; Crowley and Siegler, 1993), creativity and social intelligence (Zhang, 1997; Wei et al., 2025).

A straightforward approach is to train on a mixture of tasks. However, we observe that naive mixing often leads to performance degradation due to trade-offs among tasks. This issue arises because mixed training implicitly optimizes an **OR**-style objective, where high rewards can be achieved by excelling at only a subset of tasks. To address this limitation, we propose a **nested training framework** that transforms the implicit **OR** objective into an explicit **AND** objective (Fig. 1). In this framework, nested tasks are constructed by sequentially composing multiple sub-tasks, and the model receives maximal reward only when it performs well across all components, thereby explicitly encouraging the simultaneous acquisition of multiple abilities with more stable gradients and higher entropy. Experimental results across a broad set of ability-oriented benchmarks show that augmenting formal learning with game-based informal learning consistently improves general performance. On average, general ability increases from 38.34% to 42.43% for 1.5B models, while a substantially larger gain is observed for 7B models, rising from 42.00% to 55.84%. Moreover, greater diversity in nested game types further enhances generalization, with performance improving from 40.40% to 42.43% for 1.5B models and from 54.95% to 55.84% for 7B models.

Our contributions are summarized as follows:

- 1) **Informal learning Paradigm:** We propose a novel perspective that conceptualizes games as the fundamental environment for the informal learning of LLMs, providing a scalable, interactive environment without the need for manual annotation;

- 2) **Methodological Innovation:** We identify the "OR-style" optimization trap in naive mixed training and propose a Nested Training Framework, which transforms the objective into an explicit "AND" logic and ensures the simultaneous acquisition of diverse abilities;
- 3) **Empirical Validation:** Through extensive experiments on ability-oriented benchmarks, we demonstrate that game-based informal learning significantly enhances LLMs beyond formal learning only.

2 Related Works

2.1 Games and Large Language Models

Recent studies on LLMs and games can be broadly categorized into two research directions. The first direction, commonly referred as LLM for Games, investigates the training and evaluation of LLMs within specific game environments. Representative works examine LLM behavior in negotiation games, social deduction settings, and multi-agent text-based games, aiming to assess strategic consistency, equilibrium behavior, or task-specific performance (Fan et al., 2024; Mao et al., 2025; Bianchi et al., 2024; Guertler et al., 2025; Akata et al., 2025). These studies primarily focus on understanding or improving LLM performance within particular games, rather than enhancing general reasoning or learning capabilities across tasks. In contrast, the Game for LLM line of research treats games as structured interaction frameworks for improving broader LLM abilities. Prior works demonstrate that self-play, repeated interactions, and multi-agent game dynamics can facilitate improvements in reasoning, alignment, and strategic adaptation (Tang et al., 2025; Liu et al., 2025b; Xie et al., 2025). However, existing studies remain

limited in game diversities and provide insufficient analysis on multi-task settings.

2.2 Multi-Task RL Training in LLMs

Reinforcement learning (RL) has been shown to play a critical role in enhancing the reasoning capabilities of large language models (LLMs) (Liu et al., 2024; Wang et al., 2024; Khatri et al., 2025; Guo et al., 2025; Xu et al., 2025). A growing body of work demonstrates that RL-based training can significantly improve performance on reasoning-intensive tasks, such as mathematical problem solving (Zeng et al., 2025a; Wang et al., 2025b) and code generation (Zhao et al., 2025). More recently, several works have begun to investigate multi-task RL training for LLMs (Zeng et al., 2025b). Some studies observe that naively mixing heterogeneous tasks often leads to performance trade-offs (Wu et al., 2025). To mitigate this issue, OMNI-THINKER adopts a curriculum-based training strategy and mixed reward designs (Li et al., 2025) and AgentRL replaces the group-based advantage in GRPO with a task-aware advantage formulation (Zhang et al., 2025a).

In contrast to these approaches, which primarily address multi-task instability through curriculum design, reward shaping, or advantage reweighting, our work explores a complementary direction by reformulating multi-task optimization itself, enabling synergistic ability acquisition without relying on manual designs.

3 Method

In this section, we explore the motivation of introducing formal and informal learning and how informal learning signals can be integrated into model training through reinforcement learning.

3.1 Motivation

In educational theory, human learning is commonly categorized into three types: *formal learning*, *non-formal learning*, and *informal learning* (Coombs and Ahmed, 1974; Johnson and Majewska, 2022). Formal learning refers to institutionalized education systems with well-defined curricula, while non-formal learning is the structured education outside the standard formal education system. Both formal and non-formal learning reflect accumulated task-specific knowledge (Cattell, 1963; Horn and Cattell, 1967). In contrast, informal learning is defined as learning that arises from everyday activities

and takes place through immersion in interactive environments, where learning is conducted through practice and feedback. This contributes to solving problems that are independent of specific tasks and supports general abilities that adapt to unseen situations (Ziegler et al., 2012; Thorsen et al., 2014). Since non-formal learning shares similar characteristics with formal learning, we exclude it in this work.

Despite the central role of formal learning in current LLM training, particularly through structured tasks such as mathematics and code, educational research shows that informal learning activities play a more significant role (Fevre et al., 2001; Za et al., 2014). Informal learning has been strongly associated with the development of effective problem-solving skills in technology-rich environments (Nygren et al., 2019) and is more critical in skill development than formal training courses in workplaces (De Grip, 2024; Fevre et al., 2001). Moreover, previous studies indicate that the combination of formal and informal learning is particularly beneficial (Gerber et al., 2001). These findings suggest that informal learning is a crucial yet underexplored component to developing generalizable intelligence and the combination of formal and informal learning is necessary.

3.2 Game as Informal Learning Environments

Formal learning is characterized by structured acquisition of explicit knowledge. In LLM training, such learning paradigms are naturally instantiated by mathematics, where models are trained to solve well-defined problems under supervision. Accordingly, we formulate *Math* as the formal learning environment in our study.

Motivated by the central role of informal learning in human cognition, we consider what constitutes informal learning for LLMs. Informal learning is characterized by three key properties: it emerges from unstructured interaction, experience, and feedback; it proceeds without explicit instruction and predefined learning objectives; and it spans diverse and heterogeneous scenarios.

From this perspective, GAME-based environments provide a natural and effective abstraction for modeling informal learning. Games are inherently interaction-driven and governed by intrinsic rules that generate feedback and rewards through agent-environment dynamics. Learning signals arise directly from interaction outcomes, eliminating the need for human-annotated data. Moreover,

Category	Game	Targeted Abilities
Single-turn	<i>Matrix Games</i>	Abstract & Strategic reasoning;
Multi-turn two-player	<i>TicTacToe</i>	Long-horizon planning; Sequential decision making;
Multi-turn multi-player	<i>Who's the Spy</i>	Theory of Mind; Creative language generation

Table 1: Representative game environments and the reasoning abilities they promote.

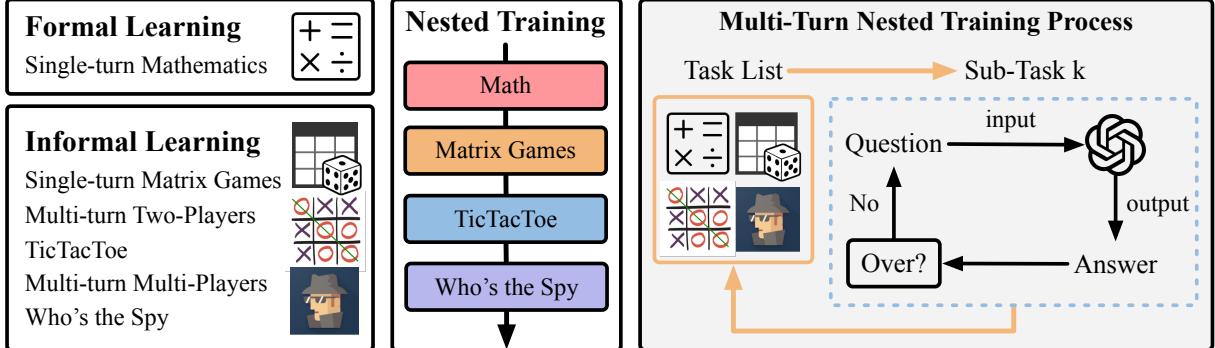


Figure 2: Overview of the proposed nested training framework with formal and informal learning tasks. The middle panel presents a high-level abstraction of the nested task structure, while the right panel details the multi-turn nested training process with iterative model-environment interactions.

games distill essential structures of real-world scenarios into controlled yet expressive settings, enabling models to explore strategies, adapt to feedback, and reason about others across a wide range of simulated scenarios.

To systematically capture the diversity of informal learning experiences, we organize game environments along the degree of social interaction complexity (single-turn, multi-turn two-player, and multi-turn multi-player settings). We design and select three representative game environments, summarized in Table 1, each targeting complementary aspects of informal learning and ability acquisition (Noda et al., 2019; Liang et al., 2025). The detailed game information is introduced in Appendix A.

3.3 Naive Mixed Training

Given the diversity of formal and informal learning tasks, which instantiate informal learning and target complementary abilities, a natural strategy is to jointly train a single model on all tasks. Such naive mixed training exposes the model to heterogeneous learning signals and is expected to facilitate the acquisition of diverse capabilities.

However, our empirical results show that directly mixing formal and informal learning tasks often leads to suboptimal outcomes, as summarized in Table 2 and Table 3. Although each task is effective in isolation, naive mixed training frequently suffers from unstable optimization and negative interference across tasks. In practice, the model tends to

over-optimize a subset of tasks while neglecting others, hindering the simultaneous improvement of diverse abilities, illustrated in Fig. 3.

From an optimization perspective, mixed training aggregates learning signals across tasks, where achieving success on *any* sub-task is sufficient to increase the overall objective. Concretely, given K sub-tasks with task-specific rewards $R_k(\tau_k)$, where τ_k denotes the trajectory for task k , mixed training optimizes the following additive objective:

$$\max_{\theta} \mathbb{E}_{\tau \sim \pi_{\theta}} \left[\sum_{k=1}^K R_k(\tau_k) \right]. \quad (1)$$

As a result, once a particular sub-task reaches high performance, its reward may dominate the gradient signal, causing gradients associated with other sub-tasks to diminish or vanish, without yielding commensurate learning gains (Wu et al., 2025).

As shown in Fig. 4, the OR-type objective allows individual tasks to converge independently, leading to low policy entropy within each task. At the same time, it does not enforce coordination across tasks, causing different tasks to alternately dominate the reward signal. Due to varying reward scales and success probabilities, the dominant optimization signal shifts across batches, resulting in highly variable gradient magnitudes. Together, these dynamics undermine optimization stability and ultimately lead to poor joint generalization.

Importantly, this limitation is intrinsic to the OR-type objective structure, which allows sub-tasks to

be optimized independently instead of encouraging a unified solution.

3.4 Nested Training Framework

To address this limitation, we propose a **Nested** training framework that constructs a composite task by hierarchically combining multiple sub-tasks, as illustrated in Fig. 2. Task success is defined by the **joint satisfaction** of all sub-tasks under a **global, order-invariant objective**, where the model attains the highest reward only when **all** sub-tasks are completed. Partial success does not saturate the objective and continues to provide informative gradients for optimization.

In this way, nested training replaces the OR-type objective in mixed training with an explicit **AND-type** success condition, formalized as

$$\max_{\theta} \mathbb{E}_{\tau \sim \pi_{\theta}} [R(\tau_1, \tau_2, \dots, \tau_K)]. \quad (2)$$

Because partial success does not saturate the AND-type objective, the policy avoids premature commitment, preserving exploration across sub-tasks and maintaining higher entropy throughout training. Meanwhile, by coupling optimization signals across all sub-tasks, the AND-type objective prevents any single task from dominating the gradient, yielding balanced and stable optimization dynamics. As shown in Fig. 4, sustained entropy and stable gradients together enable steady, coordinated improvement across tasks and result in superior joint generalization.

Importantly, nested training is neither a stricter reward scaling nor a multiplicative objective. Rather, it elevates the learning problem to a higher level: the optimization target is no longer individual task performance, but the acquisition of a **joint capability** that is sufficient to solve all sub-tasks simultaneously. Detailed information about nested training settings are in Appendix B.2.

4 Experiment

In this section, we present experimental results to evaluate the effectiveness of combining formal and informal learning, as well as the impact of the proposed nested training framework.

4.1 Environment Setup

We briefly describe the environments, training configurations, and evaluation protocols used in our experiments, with details in the Appendix.

4.1.1 Environments and Tasks

We adopt *Math* as the formal learning environment and a set of game-based environments as informal learning tasks, including *Matrix Games*, *TicTacToe*, and *Who's the Spy*. For *Math* training, we use MathLv3-5 problems from the SimpleRL-Zoo-Data, following the setup in (Zeng et al., 2025a). For game-based training, we refer to Appendix A for detailed environment descriptions and prompting strategies.

4.1.2 Training Settings

Following the RAGEN framework, we train language models using StarPO*, with trajectory-based reinforcement learning formulation using GRPO. We evaluate two model scales, Qwen2.5-1.5B-Instruct and Qwen2.5-7B-Instruct (Yang et al., 2024; Team, 2024), as base models. In two-player and multi-player games, we employ Qwen3-14B as opponents (Team, 2025). Full training parameters and implementation are reported in Appendix B.

We report three main experimental settings: single-task training, multi-task mixed training, and multi-task nested training. For multi-task configurations, we investigate progressive combinations of formal and informal learning and denote the setups as $F + I_k$, where k indicates the number of informal learning components included in training, reflecting the depth and complexity of informal learning. Specifically, I_1 , I_2 , and I_3 correspond to *Matrix Games*, *Matrix Games* combined with *TicTacToe*, and the full combination including *Matrix Games*, *TicTacToe*, and *Who's the Spy*, respectively.

4.1.3 Evaluation Metrics

We evaluate in-domain performance on the training tasks, including the MATH500 benchmark (Lightman et al., 2023), *Matrix Games*, *TicTacToe*, and *Who's the Spy*. For games with opponents, including *TicTacToe* and *Who's the Spy*, we report the average success rate against Gemini-2.5-Flash in 100 rounds (Comanici et al., 2025). For general and diverse abilities, we choose MMLU and MMLU-Pro for multi-domain reasoning (Hendrycks et al., 2021a,b), CommonGen for creative language generation (Lin et al., 2020) and SocialIQA for social reasoning (Sap et al., 2019). In CommonGen tasks, we use GPT-4o (Hurst et al., 2024) to compare the generated outputs against ground truth references, counting better and semantically equivalent generations (ties) as success. The details are in Appendix C.

Setting	Model	MATH	Matrix	TicTacToe	Spy	MMLU	MMLU-Pro	Common	Social	Avg.
Base	Qwen2.5-1.5B	17.20	7.00	1.00	2.00	37.87	13.49	7.79	27.64	27.10
Formal	<i>Math</i>	43.20	21.00	4.00	9.00	51.38	20.97	15.08	65.92	38.34
Informal	<i>Matrix Games</i>	19.20	44.00	34.00	14.00	43.79	18.00	9.55	64.12	33.87
	<i>TicTacToe</i>	22.40	32.00	75.00	21.00	47.89	16.45	11.56	64.12	35.00
	<i>Who's the Spy</i>	19.20	21.00	0.00	33.00	43.85	15.68	20.85	60.70	35.27
F + I ₁	mixed	37.80	29.00	0.00	16.00	45.19	17.00	12.81	59.11	33.53
	nested	40.00	65.00	8.00	24.00	52.94	21.03	21.11	66.53	40.40
F + I ₂	mixed	28.00	30.00	50.00	20.00	51.35	19.47	17.84	65.30	38.49
	nested	16.20	20.00	16.00	26.00	53.08	20.20	20.85	67.09	40.31
F + I ₃	mixed	34.60	55.00	34.00	37.00	53.55	21.19	24.37	67.60	41.68
	nested	22.20	57.00	46.00	35.00	53.27	20.58	28.14	67.71	42.43

Table 2: Performance of single-tasks, mixed multi-tasks, and nested multi-tasks training for base model Qwen2.5-1.5B-Instruct, where F + I_k denotes the combination of formal learning and *k* informal learning tasks. Purple shading highlights the in-domain tasks for each setting.

Setting	Model	MATH	Matrix	TicTacToe	Spy	MMLU	MMLU-Pro	Common	Social	Avg.
Base	Qwen2.5-7B	54.60	40.00	40.00	25.00	71.43	40.17	26.38	75.44	53.36
Formal	<i>Math</i>	58.40	39.00	31.00	18.00	66.51	32.45	35.18	75.90	42.00
Informal	<i>Matrix Games</i>	58.20	64.00	23.00	27.00	73.22	44.51	26.13	76.20	55.02
	<i>TicTacToe</i>	49.80	57.00	78.00	33.00	73.37	46.35	28.89	76.41	56.26
	<i>Who's the Spy</i>	55.60	43.00	38.00	37.00	71.33	41.17	28.39	74.92	53.95
F + I ₁	mixed	58.20	52.00	24.00	21.00	67.55	35.21	27.89	73.34	51.00
	nested	57.40	43.00	33.00	33.00	68.33	35.56	41.46	74.46	54.95
F + I ₂	mixed	50.00	63.00	45.00	9.00	66.80	37.81	36.68	72.21	53.38
	nested	55.60	40.00	33.00	40.00	71.31	44.00	28.89	74.41	54.65
F + I ₃	mixed	43.40	49.00	36.00	62.00	68.64	36.87	35.68	74.41	53.90
	nested	52.20	47.00	47.00	45.00	70.50	40.86	35.43	76.56	55.84

Table 3: Performance of single-tasks, mixed multi-tasks, and nested multi-tasks training for base model Qwen2.5-7B-Instruct, where F + I_k denotes the combination of formal learning and *k* informal learning tasks.

Notation. We use MATH, Matrix, Spy, Common and Social to denote MATH500, *Matrix Games*, *Who's the Spy*, CommonGen and SocialIQA respectively. Avg denotes the average performance over general ability benchmarks, including MMLU, MMLU-Pro, CommonGen and SocialIQA. We report math performance using accuracy, game performance using success rates and general-ability performance using accuracy; all values are reported in percentage form, with the percentage symbol omitted for brevity in the tables.

4.2 Main Results

Table 2 and 3 summarize the main experimental results across two model scales. We analyze these results from two perspectives: the role of formal versus informal learning and the effectiveness of nested training.

Formal vs. Informal Learning. We first examine whether distinguishing formal and informal learning is empirically meaningful. Results from single-task training show that both paradigms contribute to general ability improvement, but in complementary ways. Formal learning primarily benefits structured reasoning tasks (MMLU and MMLU-Pro), while informal learning yields stronger gains on creative writing and socially grounded benchmarks (CommonGen and SocialIQA), especially in 7B base model. These observations validate the necessity of separating formal and informal learning signals and are consistent with insights from cognitive science.

Effectiveness of Nested Training. Although both formal and informal learning are beneficial when applied in isolation, naively mixing their

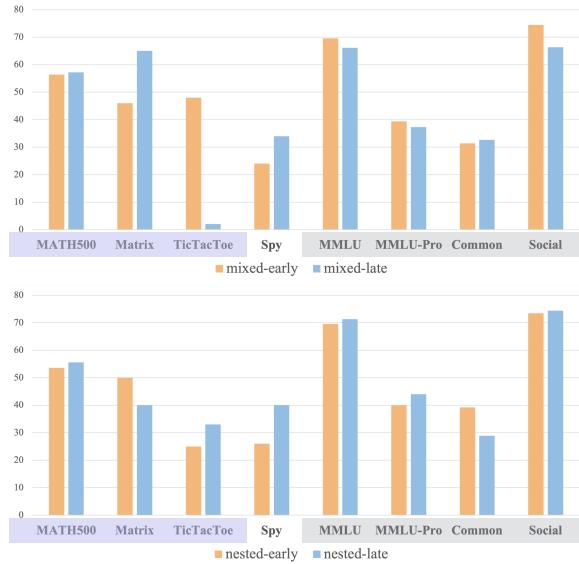


Figure 3: Comparison between mixed and nested training in $F + I_2$ setting with 7B base model. Vertical axis denotes the performance, purple color denotes the ID tasks and gray color denotes the general abilities.

learning signals does not consistently translate into performance gains. In contrast, nested training substantially alleviates performance degradation on general abilities across **all** settings. The effect is particularly pronounced in the $F + I_1$ setting with the 1.5B base model, where nested training yields a **6.87% absolute improvement**.

To further analyze this behavior, we compare the training dynamics of mixed and nested training in Fig. 3 and Fig. 4. Although mixed training achieves higher in-domain performance at early stages, it gradually collapses toward specific tasks (e.g., *Matrix Games*), accompanied by unstable gradients and rapidly decreasing entropy, which ultimately degrades generalization performance. As a result, mixed training may exhibit stronger in-domain results in the table, while nested training prioritizes balanced optimization across tasks. In contrast, nested training maintains stable gradients and higher entropy throughout training, leading to steady and consistent improvements across most abilities. These results demonstrate the superior optimization stability and overall effectiveness of the proposed nested training framework.

4.3 Case Study

To further illustrate the effectiveness of informal learning, we present a compact case study in Fig. 5, which summarizes two representative examples, with detailed comparisons in the Appendix (Fig. 7

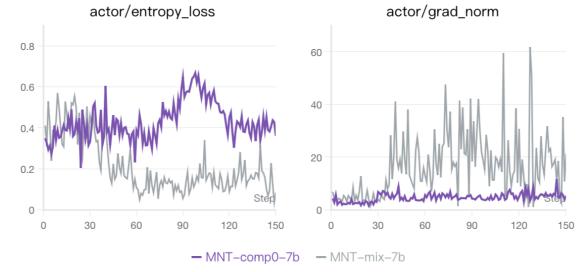


Figure 4: Comparison between mixed and nested training in $F + I_2$ setting with 7B base model.

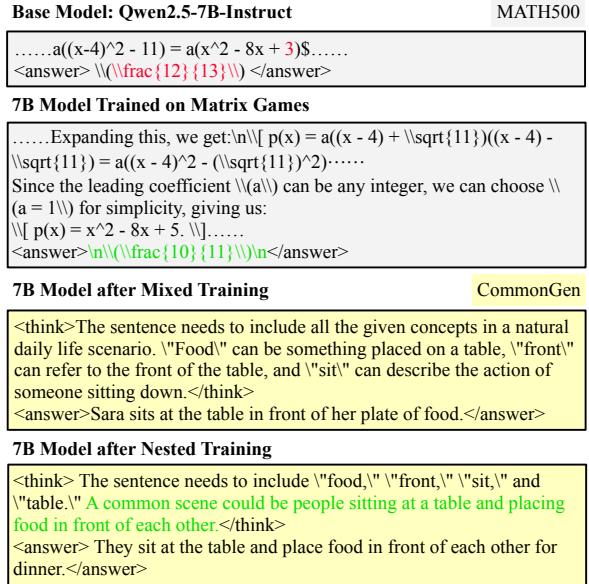


Figure 5: Case Studies on MATH500 and CommonGen.

and 8). On the MATH500 benchmark, the base 7B model follows the correct high-level reasoning but makes a subtle arithmetic error, leading to an incorrect result, whereas the *Matrix Games*-trained model maintains a more explicit and verifiable derivation. Moreover, it chooses $a = 1$ creatively to simplify the calculation. On the CommonGen benchmark, while mixed training on $F + I_2$ setting produces a semantically reasonable sentence, nested training encourages deeper semantic integration by first validating the given concepts and then constructing a coherent and detailed scene. Together, these cases demonstrate that informal learning with the nested training framework promotes more explicit, robust, and creative reasoning across both mathematical and generative tasks.

4.4 Ablation Study

We conduct a series of ablation studies to further analyze the generality, necessity, and sensitivity of the proposed nested training framework.

Setting	Model	MATH	Matrix	TicTacToe	Spy	MMLU	MMLU-Pro	Common	Social	Avg.
F + I ₂	mixed	28.00	30.00	50.00	20.00	51.35	19.47	17.84	65.30	38.49
	nested	16.20	20.00	16.00	26.00	53.08	20.20	20.85	67.09	40.31
I ₂	mixed	9.40	49.00	51.00	26.00	39.57	15.78	17.34	59.98	33.17
	nested	20.20	24.00	22.00	9.00	43.80	16.25	17.09	60.70	34.46

Table 4: Effect of formal learning and nested training under the informal-only setting I₂ with 1.5B base model.

4.4.1 Nested Training Beyond Formal+Informal

To examine whether the nested training framework generalizes beyond the combination of formal and informal learning, we evaluate it under an informal-learning-only setting, denoted as I₂, which consists of *Matrix Games* and *TicTacToe*. As shown in Table 4, nested training remains effective in this setting, improving MMLU performance from 39.57% to 43.80% and increasing the average general ability score from 33.17% to 34.46%. These results demonstrate that the nested framework provides stable gains even when applied solely to informal learning environments, indicating its robustness beyond the formal-informal combination.

4.4.2 Necessity of Formal vs Informal Learning

The main experiments already show that combining formal and informal learning yields stronger general abilities than formal learning alone. Here, we further examine whether informal learning is sufficient, as assumed in many game-centric LLM training approaches.

Using the same I₂ configuration on the 1.5B base model, we compare informal-only training against the formal+informal(F + I₂) setting. As shown in Table 4, models trained solely on informal learning tasks consistently underperform those trained with both formal and informal learning across most benchmarks, under both mixed and nested training. In particular, removing *Math* signals leads to a performance drop of 9.28% on MMLU and 5.85% on the average general ability score. These results highlight the critical role of formal learning, suggesting that it cannot be fully replaced by informal learning alone.

4.4.3 Opponent Sensitivity in Multi-player Games

We study the effect of opponent choice by training the 1.5B model against different opponents, as shown in Table 5 (Appendix). We find that

Qwen3-14B outperforms Gemini-2.5-Flash in direct evaluations, achieving success rates of 66% in *TicTacToe* and 52% in *Who’s the Spy*. Training against the Qwen3-14B opponent yields better *TicTacToe* performance, aligning with the intuition that stronger opponents encourage the learning of more robust strategies. Moreover, open-sourced Qwen3-14B can be deployed locally with lower inference latency. Therefore, it is adopted as the default opponent in all experiments.

4.4.4 Order Sensitivity of Nested Training

From the nested training reward definition, the nested objective is theoretically invariant to the execution order of sub-tasks. To empirically validate this property, we conduct an order-sensitivity ablation under the F + I₁ setting using the 7B base model. Specifically, we compare two nested configurations: *Math* → *Matrix Games* and *Matrix Games* → *Math*. As shown in Table 6 (Appendix), the two configurations yield comparable performance on average general abilities, between 54.95% and 54.59%. These results indicate that the proposed nested training framework is largely robust to the ordering of sub-tasks, confirming that its effectiveness does not rely on a specific order.

5 Conclusion

Inspired by theories in cognitive science, we model formal learning as structured *Math* reasoning tasks and informal learning as interactive, game-based environments. We design three representative game settings: single-turn *Matrix Games*, multi-turn two-player *TicTacToe*, and multi-turn multi-player *Who’s the Spy*. While a naive mixed training strategy leads to performance degradation, we show that the nested training framework enables effective integrations. Extensive experiments demonstrate that the combination of formal and informal learning is necessary and compared to mixed training, nested training improves both generalization and training stability across diverse settings, resulting in LLMs with broader abilities.

6 Limitations

This work considers a limited set of formal and informal learning environments, and the game designs represent only a subset of possible interactive settings. While the nested framework generalizes beyond the studied combinations, its effectiveness in more complex or real-world interactive environments remains to be explored. Moreover, our experiments use fixed opponent models in multi-agent games, and extending to self-evolve settings or adaptive opponents is left for future work.

References

- Elif Akata, Lion Schulz, Julian Coda-Forno, Seong Joon Oh, Matthias Bethge, and Eric Schulz. 2025. Playing repeated games with large language models. *Nature Human Behaviour*, pages 1–11.
- Robert J Aumann and Sergiu Hart. 1992. *Handbook of game theory with economic applications*, volume 2. Elsevier.
- Federico Bianchi, Patrick John Chia, Mert Yuksekgonul, Jacopo Tagliabue, Dan Jurafsky, and James Zou. 2024. How well can llms negotiate? negotiationarena platform and analysis. *arXiv preprint arXiv:2402.05863*.
- Maureen Callanan, Christi Cervantes, and Molly Loomis. 2011. Informal learning. *Wiley Interdisciplinary Reviews: Cognitive Science*, 2(6):646–655.
- Raymond B Cattell. 1963. Theory of fluid and crystallized intelligence: A critical experiment. *Journal of educational psychology*, 54(1):1.
- Luxin Chen, Jinming Gu, Liankai Huang, Wenhao Huang, Zhicheng Jiang, Allan Jie, Xiaoran Jin, Xing Jin, Chenggang Li, Kaijing Ma, and 1 others. 2025. Seed-prover: Deep and broad reasoning for automated theorem proving. *arXiv preprint arXiv:2507.23726*.
- Gheorghe Comanici, Eric Bieber, Mike Schaeckermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.
- Philip Hall Coombs and Manzoor Ahmed. 1974. *Attacking rural poverty. How nonformal education can help*.
- Kevin Crowley and Robert S Siegler. 1993. Flexible strategy use in young children’s tic-tac-toe. *Cognitive Science*, 17(4):531–561.
- Andries De Grip. 2024. The importance of informal learning at work. *IZA World of Labor*.
- Prajit K Dutta. 1999. *Strategies and games: theory and practice*. MIT press.
- Peter Edwards, Lisa Sharma-Wallace, Anita Wreford, Lania Holt, Nicholas A Cradock-Henry, Stephen Flood, and Sandra J Velarde. 2019. Tools for adaptive governance for complex social-ecological systems: a review of role-playing-games as serious games at the community-policy interface. *Environmental Research Letters*, 14(11):113002.
- Caoyun Fan, Jindou Chen, Yaohui Jin, and Hao He. 2024. Can large language models serve as rational players in game theory? a systematic analysis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17960–17967.
- Ralph Fevre, S Gorad, and Gareth Rees. 2001. Necessary and unnecessary learning: the acquisition of knowledge and “skills” in and outside employment in south wales in the 20 th century. *The necessity of informal learning*.
- Brian L Gerber, Anne ML Cavallo, and Edmund A Marek. 2001. Relationships among informal learning environments, teaching procedures and scientific reasoning ability. *International Journal of Science Education*, 23(5):535–549.
- Leon Guertler, Bobby Cheng, Simon Yu, Bo Liu, Leshem Choshen, and Cheston Tan. 2025. Textarena. *arXiv preprint arXiv:2504.11442*.
- Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shiron Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.
- Dan Hendrycks, Collin Burns, Steven Basart, Andrew Critch, Jerry Li, Dawn Song, and Jacob Steinhardt. 2021a. Aligning ai with shared human values. *Proceedings of the International Conference on Learning Representations (ICLR)*.
- Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021b. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*.
- John L Horn and Raymond B Cattell. 1967. Age differences in fluid and crystallized intelligence. *Acta Psychologica*, 26:107–129.
- Wenyue Hua, Ollie Liu, Lingyao Li, Alfonso Amayuelas, Julie Chen, Lucas Jiang, Mingyu Jin, Lizhou Fan, Fei Sun, William Wang, and 1 others. 2024. Game-theoretic llm: Agent workflow for negotiation games. *arXiv preprint arXiv:2411.05990*.
- Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

- Judith E Innes and David E Booher. 1999. Consensus building as role playing and bricolage: Toward a theory of collaborative planning. *Journal of the american planning association*, 65(1):9–26.
- Martin Johnson and Dominika Majewska. 2022. Formal, non-formal, and informal learning: What are they, and how can we research them? research report. *Cambridge University Press & Assessment*.
- Devvrit Khatri, Lovish Madaan, Rishabh Tiwari, Rachit Bansal, Sai Surya Duvvuri, Manzil Zaheer, Inderjit S Dhillon, David Brandfonbrener, and Rishabh Agarwal. 2025. The art of scaling reinforcement learning compute for llms. *arXiv preprint arXiv:2510.13786*.
- Willy Christian Kriz, Junkichi Sugiura, and Toshiko Kikkawa. 2022. Gaming simulation: Terminology and fundamentals. In *Gaming as a cultural commons: Risks, challenges, and opportunities*, pages 3–23. Springer.
- Derek Li, Jiaming Zhou, Amirreza Kazemi, Qianyi Sun, Abbas Ghaddar, Liheng Ma, Yu Luo, Dong Li, Jianye HAO, and Yingxue Zhang. 2025. Omni-thinker: Scaling cross-domain generalization in llms via multi-task rl with hybrid rewards. In *2nd AI for Math Workshop@ ICML 2025*.
- Dayong Liang, Xiao-Yong Wei, and Changmeng Zheng. 2025. Multi-agent undercover gaming: Hallucination removal via counterfactual test for multimodal reasoning. *arXiv preprint arXiv:2511.11182*.
- Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. Let’s verify step by step. *arXiv preprint arXiv:2305.20050*.
- Bill Yuchen Lin, Wangchunshu Zhou, Ming Shen, Pei Zhou, Chandra Bhagavatula, Yejin Choi, and Xiang Ren. 2020. CommonGen: A constrained text generation challenge for generative commonsense reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1823–1840, Online. Association for Computational Linguistics.
- Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, and 1 others. 2024. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.
- Aixin Liu, Aoxue Mei, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, and 1 others. 2025a. Deepseek-v3. 2: Pushing the frontier of open large language models. *arXiv preprint arXiv:2512.02556*.
- Bo Liu, Leon Guertler, Simon Yu, Zichen Liu, Penghui Qi, Daniel Balcells, Mickel Liu, Cheston Tan, Weiyan Shi, Min Lin, and 1 others. 2025b. Spirral: Self-play on zero-sum games incentivizes reasoning via multi-agent multi-turn reinforcement learning. *arXiv preprint arXiv:2506.24119*.
- Keliang Liu, Dingkang Yang, Ziyun Qian, Weijie Yin, Yuchi Wang, Hongsheng Li, Jun Liu, Peng Zhai, Yang Liu, and Lihua Zhang. 2025c. Reinforcement learning meets large language models: A survey of advancements and applications across the llm lifecycle. *arXiv preprint arXiv:2509.16679*.
- William F Lucas. 1981. *Game theory and its applications*, volume 24. American Mathematical Soc.
- Shaoguang Mao, Yuzhe Cai, Yan Xia, Wenshan Wu, Xun Wang, Fengyi Wang, Qiang Guan, Tao Ge, and Furu Wei. 2025. Olympics: Llm agents meet game theory. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 2845–2866.
- Prakamya Mishra, Jiang Liu, Jialian Wu, Xiaodong Yu, Zicheng Liu, and Emad Barsoum. 2025. Ttt-bench: A benchmark for evaluating reasoning ability with simple and novel tic-tac-toe-style games. *arXiv preprint arXiv:2506.10209*.
- Shota Noda, Kentaro Shirotsuki, and Mutsuhiro Nakao. 2019. The effectiveness of intervention with board games: a systematic review. *BioPsychoSocial medicine*, 13(1):22.
- Henrik Nygren, Kari Nissinen, Raija Hämäläinen, and Bram De Wever. 2019. Lifelong learning: Formal, non-formal and informal learning in the context of the use of problem-solving skills in technology-rich environments. *British Journal of Educational Technology*, 50(4):1759–1770.
- Bill Roungas, Femke Bekius, and Sebastiaan Meijer. 2019. The game between game theory and gaming simulations: design choices. *Simulation & Gaming*, 50(2):180–201.
- Maarten Sap, Hannah Rashkin, Derek Chen, Ronan LeBras, and Yejin Choi. 2019. Social iqai: Commonsense reasoning about social interactions. In *EMNLP*.
- Sylvia Scribner and Michael Cole. 1973. Cognitive consequences of formal and informal education: New accommodations are needed between school-based learning and learning experiences of everyday life. *Science*, 182(4112):553–559.
- ByteDance Seed, Yuyu Zhang, Jing Su, Yifan Sun, Chenguang Xi, Xia Xiao, Shen Zheng, Anxiang Zhang, Kaibo Liu, Daoguang Zan, and 1 others. 2025. Seed-coder: Let the code model curate data for itself. *arXiv preprint arXiv:2506.03524*.
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

- Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2024. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv:2409.19256*.
- Xiaohang Tang, Sangwoong Yoon, Seongho Son, Huizhuo Yuan, Quanquan Gu, and Ilija Bogunovic. 2025. Game-theoretic regularized self-play alignment of large language models. *arXiv preprint arXiv:2503.00030*.
- Qwen Team. 2024. *Qwen2.5: A party of foundation models*.
- Qwen Team. 2025. *Qwen3 technical report. Preprint, arXiv:2505.09388*.
- Cecilia Thorsen, Jan-Eric Gustafsson, and Christina Cliffordson. 2014. The influence of fluid and crystallized intelligence on the development of knowledge and skills. *British Journal of Educational Psychology*, 84(4):556–570.
- Ning Wang, Bingkun Yao, Jie Zhou, Yuchen Hu, Xi Wang, Zhe Jiang, and Nan Guan. 2025a. Large language model for verilog generation with code-structure-guided reinforcement learning. In *2025 IEEE International Conference on LLM-Aided Design (ICLAD)*, pages 164–170. IEEE.
- Shuhe Wang, Shengyu Zhang, Jie Zhang, Runyi Hu, Xiaoya Li, Tianwei Zhang, Jiwei Li, Fei Wu, Guoyin Wang, and Eduard Hovy. 2024. Reinforcement learning enhanced llms: A survey. *arXiv preprint arXiv:2412.10400*.
- Yiping Wang, Qing Yang, Zhiyuan Zeng, Liliang Ren, Liyuan Liu, Baolin Peng, Hao Cheng, Xuehai He, Kuan Wang, Jianfeng Gao, and 1 others. 2025b. Reinforcement learning for reasoning in large language models with one training example. *arXiv preprint arXiv:2504.20571*.
- Zihan Wang, Kangrui Wang, Qineng Wang, Pingyue Zhang, Linjie Li, Zhengyuan Yang, Xing Jin, Kefan Yu, Minh Nhat Nguyen, Licheng Liu, and 1 others. 2025c. Ragen: Understanding self-evolution in llm agents via multi-turn reinforcement learning. *arXiv preprint arXiv:2504.20073*.
- Chentian Wei, Jiewei Chen, and Jinzhu Xu. 2025. Exploring large language models for word games: Who is the spy? *arXiv preprint arXiv:2503.15235*.
- Runzhe Wu, Ankur Samanta, Ayush Jain, Scott Fujimoto, Jeongyeol Kwon, Ben Kretzu, Youliang Yu, Kaveh Hassani, Boris Vidolov, and Yonathan Efroni. 2025. Imbalanced gradients in rl post-training of multi-task llms. *arXiv preprint arXiv:2510.19178*.
- Yunfei Xie, Yinsong Ma, Shiyi Lan, Alan Yuille, Junfei Xiao, and Chen Wei. 2025. Play to generalize: Learning to reason through game play. *arXiv preprint arXiv:2506.08011*.
- Fengli Xu, Qianyue Hao, Zefang Zong, Jingwei Wang, Yunke Zhang, Jingyi Wang, Xiaochong Lan, Jiahui Gong, Tianjian Ouyang, Fanjin Meng, and 1 others. 2025. Towards large reasoning models: A survey of reinforced reasoning with large language models. *arXiv preprint arXiv:2501.09686*.
- An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, and 40 others. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.
- Jian Yang, Wei Zhang, Shark Liu, Jiajun Wu, Shawn Guo, and Yizhi Li. 2025. From code foundation models to agents and applications: A practical guide to code intelligence. *arXiv preprint arXiv:2511.18538*.
- Qiyi Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, and 1 others. 2025. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*.
- Stefano Za, Paolo Spagnoletti, and Andrea North-Samardzic. 2014. Organisational learning as an emerging process: The generative role of digital tools in informal learning practices. *British Journal of Educational Technology*, 45(6):1023–1035.
- Weihao Zeng, Yuzhen Huang, Qian Liu, Wei Liu, Keqing He, Zejun Ma, and Junxian He. 2025a. Simplerl-zoo: Investigating and taming zero reinforcement learning for open base models in the wild. *arXiv preprint arXiv:2503.18892*.
- Zhiyuan Zeng, Hamish Ivison, Yiping Wang, Lifan Yuan, Shuyue Stella Li, Zhuorui Ye, Siting Li, Jacqueline He, Runlong Zhou, Tong Chen, and 1 others. 2025b. Rlve: Scaling up reinforcement learning for language models with adaptive verifiable environments. *arXiv preprint arXiv:2511.07317*.
- Hanchen Zhang, Xiao Liu, Bowen Lv, Xueqiao Sun, Bohao Jing, Iat Long Iong, Zhenyu Hou, Zehan Qi, Hanyu Lai, Yifan Xu, and 1 others. 2025a. Agentrl: Scaling agentic reinforcement learning with a multi-turn, multi-task framework. *arXiv preprint arXiv:2510.04206*.
- Jiajie Zhang. 1997. The nature of external representations in problem solving. *Cognitive science*, 21(2):179–217.
- Ziyin Zhang, Jiahao Xu, Zhiwei He, Tian Liang, Qizuzhi Liu, Yansi Li, Linfeng Song, Zhenwen Liang, Zhuosheng Zhang, Rui Wang, and 1 others. 2025b. Deeptheorem: Advancing llm reasoning for theorem proving through natural language and reinforcement learning. *arXiv preprint arXiv:2505.23754*.
- Andrew Zhao, Yiran Wu, Yang Yue, Tong Wu, Quentin Xu, Matthieu Lin, Shenzhi Wang, Qingyun Wu, Zilong Zheng, and Gao Huang. 2025. Absolute zero:

Reinforced self-play reasoning with zero data. *arXiv preprint arXiv:2505.03335*.

Matthias Ziegler, Erik Danay, Moritz Heene, Jens Asendorpf, and Markus Büchner. 2012. Openness, fluid intelligence, and crystallized intelligence: Toward an integrative model. *Journal of Research in Personality*, 46(2):173–183.

A Game Rules

We categorize games into three types according to their interaction structure: single-turn games, multi-turn two-player games, and multi-turn multi-player games. For each category, we select a representative game, namely *Matrix Games*, *TicTacToe*, and *Who’s the Spy*, respectively. Following the terminology in VeRL (Sheng et al., 2024), we adopt the concept of **multi-turn** rather than **multi-step**. Here, a **turn** is defined as one complete interaction round in which the trained LLM is queried to produce an action or decision, contributing to the overall game trajectory, while multi-turn refers to a game trajectory with multiple turns. For example, *Who’s the Spy* is considered a three-turn game, as each player, as well as the trained LLM, participates in three major interaction rounds: two description turns and one final voting turn. Each turn corresponds to a distinct LLM query, and the sequence of these turns together constitutes a full trajectory.

A.1 Matrix Games

Matrix Games are single-turn strategic reasoning games. We adopt a set of classic matrix games to train LLMs in strategic reasoning, specifically their ability to infer opponents’ actions and optimize their own policies toward Nash Equilibrium. In these games, the model must select an action based solely on an abstract payoff table, without access to domain-specific semantics or external knowledge. This setting encourages *abstract reasoning* by requiring the model to interpret symbolic payoffs, compare outcomes across action pairs, and reason about best responses under different opponent choices. At the same time, it fosters *strategic reasoning*: since rewards depend on both players’ decisions, the model must anticipate the opponent’s likely move, consider mutual incentives, and choose actions that remain robust under strategic interaction, rather than maximizing immediate payoff in isolation. Through reinforcement learning over repeated interactions, the model is trained to align its action selection with

equilibrium-consistent behavior, learning to balance self-interest with opponent-aware reasoning.

Following (Hua et al., 2024), we select a collection of well-studied matrix games that are verified to admit Nash Equilibrium solutions, including Prisoner’s Dilemma, Battle of the Sexes, Game of Chicken, Stag Hunt, Radio Station, IESDS, Duopolistic Game, GAME, and Weakly Dominated Game. The corresponding payoff matrices are illustrated in Fig. 6. To prevent the model from overfitting to a fixed numerical scale or specific prompt format, we apply random transformations to the payoff matrices during training. Specifically, each matrix is randomly multiplied by -1 or left unchanged, and an offset of ± 100 is added to all entries. These transformations preserve the strategic structure and equilibrium properties of the games while encouraging the model to focus on relative payoffs and strategic relationships rather than absolute values. In addition, we design multiple instruction prompt templates to present matrix games under diverse linguistic and contextual formulations. This further improves robustness to prompt variations and discourages reliance on superficial patterns. An illustrative example of such a prompt template is shown below:

```
{role}.

Rows = Player 1's actions [{  
    p1_actions_list}]; Columns =  
    Player 2's actions [{  
    p2_actions_list}].  
  
### P1's payoff  
{p1_payoff_table}  
  
### P2's payoff  
{p2_payoff_table}  
  
{instr}
```

A.2 TicTacToe

TicTacToe is a multi-turn two-player board game in which players alternately place symbols (e.g., X and O) on a 3×3 grid, with the objective of forming a straight line of three identical symbols horizontally, vertically, or diagonally. Successful play requires each player to anticipate the opponent’s potential actions and strategically plan subsequent moves, thereby encouraging the model to reason about others’ intentions and future behaviors. In addition, *TicTacToe* imposes strict constraints on valid actions: only unoccupied grid positions can

a. Prisoner's Dilemma	b. Battle of Sexes	c. Game of Chicken	d. Stag Hunt																																																																																																										
<table border="1" style="border-collapse: collapse; width: 100%;"> <thead> <tr> <th></th><th>1</th><th>2</th><th></th></tr> <tr> <th>1</th><td>(3, 3)</td><td>(0, 5)</td><td>(2, 1)</td></tr> <tr> <th>2</th><td>(5, 0)</td><td>(1, 1)</td><td>(0, 0)</td></tr> </thead> <tbody> <tr> <td></td><td></td><td></td><td>(0, 0)</td></tr> <tr> <td></td><td></td><td></td><td>(1, 2)</td></tr> </tbody> </table>		1	2		1	(3, 3)	(0, 5)	(2, 1)	2	(5, 0)	(1, 1)	(0, 0)				(0, 0)				(1, 2)	<table border="1" style="border-collapse: collapse; width: 100%;"> <thead> <tr> <th></th><th>1</th><th>2</th><th></th></tr> <tr> <th>1</th><td>(2, 1)</td><td>(0, 0)</td><td>(-10, -10)</td></tr> <tr> <th>2</th><td>(0, 0)</td><td>(1, 2)</td><td>(1, -1)</td></tr> </thead> <tbody> <tr> <td></td><td></td><td></td><td>(0, 0)</td></tr> <tr> <td></td><td></td><td></td><td>(-1, 1)</td></tr> </tbody> </table>		1	2		1	(2, 1)	(0, 0)	(-10, -10)	2	(0, 0)	(1, 2)	(1, -1)				(0, 0)				(-1, 1)	<table border="1" style="border-collapse: collapse; width: 100%;"> <thead> <tr> <th></th><th>1</th><th>2</th><th></th></tr> <tr> <th>1</th><td>(3, 3)</td><td>(0, 1)</td><td>(3, 3)</td></tr> <tr> <th>2</th><td>(1, 0)</td><td>(1, 1)</td><td>(1, 0)</td></tr> </thead> <tbody> <tr> <td></td><td></td><td></td><td>(0, 1)</td></tr> </tbody> </table>		1	2		1	(3, 3)	(0, 1)	(3, 3)	2	(1, 0)	(1, 1)	(1, 0)				(0, 1)																																																			
	1	2																																																																																																											
1	(3, 3)	(0, 5)	(2, 1)																																																																																																										
2	(5, 0)	(1, 1)	(0, 0)																																																																																																										
			(0, 0)																																																																																																										
			(1, 2)																																																																																																										
	1	2																																																																																																											
1	(2, 1)	(0, 0)	(-10, -10)																																																																																																										
2	(0, 0)	(1, 2)	(1, -1)																																																																																																										
			(0, 0)																																																																																																										
			(-1, 1)																																																																																																										
	1	2																																																																																																											
1	(3, 3)	(0, 1)	(3, 3)																																																																																																										
2	(1, 0)	(1, 1)	(1, 0)																																																																																																										
			(0, 1)																																																																																																										
e. Radio Station	f. IESDS	g. Weakly Dominated Game																																																																																																											
<table border="1" style="border-collapse: collapse; width: 100%;"> <thead> <tr> <th></th><th>1</th><th>2</th><th>3</th><th></th></tr> <tr> <th>1</th><td>(25, 25)</td><td>(50, 30)</td><td>(50, 20)</td><td>(13, 3)</td></tr> <tr> <th>2</th><td>(30, 50)</td><td>(15, 15)</td><td>(30, 20)</td><td>(4, 1)</td></tr> <tr> <th>3</th><td>(20, 50)</td><td>(20, 30)</td><td>(10, 10)</td><td>(-1, 2)</td></tr> </thead> <tbody> <tr> <td></td><td></td><td></td><td></td><td>(2, 3)</td></tr> <tr> <td></td><td></td><td></td><td></td><td>(8, -1)</td></tr> </tbody> </table>		1	2	3		1	(25, 25)	(50, 30)	(50, 20)	(13, 3)	2	(30, 50)	(15, 15)	(30, 20)	(4, 1)	3	(20, 50)	(20, 30)	(10, 10)	(-1, 2)					(2, 3)					(8, -1)	<table border="1" style="border-collapse: collapse; width: 100%;"> <thead> <tr> <th></th><th>1</th><th>2</th><th>3</th><th></th></tr> <tr> <th>1</th><td>(13, 3)</td><td>(1, 4)</td><td>(7, 3)</td><td>(5, 1)</td></tr> <tr> <th>2</th><td>(4, 1)</td><td>(3, 3)</td><td>(6, 2)</td><td>(6, 0)</td></tr> <tr> <th>3</th><td>(-1, 2)</td><td>(2, 3)</td><td>(8, -1)</td><td>(6, 4)</td></tr> </thead> <tbody> <tr> <td></td><td></td><td></td><td></td><td>(4, 1)</td></tr> </tbody> </table>		1	2	3		1	(13, 3)	(1, 4)	(7, 3)	(5, 1)	2	(4, 1)	(3, 3)	(6, 2)	(6, 0)	3	(-1, 2)	(2, 3)	(8, -1)	(6, 4)					(4, 1)	<table border="1" style="border-collapse: collapse; width: 100%;"> <thead> <tr> <th></th><th>1</th><th>2</th><th></th></tr> <tr> <th>1</th><td>(5, 1)</td><td>(4, 0)</td><td>(5, 1)</td></tr> <tr> <th>2</th><td>(6, 0)</td><td>(3, 1)</td><td>(6, 0)</td></tr> <tr> <th>3</th><td>(6, 4)</td><td>(4, 1)</td><td>(6, 4)</td></tr> </thead> <tbody> <tr> <td></td><td></td><td></td><td>(4, 1)</td></tr> </tbody> </table>		1	2		1	(5, 1)	(4, 0)	(5, 1)	2	(6, 0)	(3, 1)	(6, 0)	3	(6, 4)	(4, 1)	(6, 4)				(4, 1)																																
	1	2	3																																																																																																										
1	(25, 25)	(50, 30)	(50, 20)	(13, 3)																																																																																																									
2	(30, 50)	(15, 15)	(30, 20)	(4, 1)																																																																																																									
3	(20, 50)	(20, 30)	(10, 10)	(-1, 2)																																																																																																									
				(2, 3)																																																																																																									
				(8, -1)																																																																																																									
	1	2	3																																																																																																										
1	(13, 3)	(1, 4)	(7, 3)	(5, 1)																																																																																																									
2	(4, 1)	(3, 3)	(6, 2)	(6, 0)																																																																																																									
3	(-1, 2)	(2, 3)	(8, -1)	(6, 4)																																																																																																									
				(4, 1)																																																																																																									
	1	2																																																																																																											
1	(5, 1)	(4, 0)	(5, 1)																																																																																																										
2	(6, 0)	(3, 1)	(6, 0)																																																																																																										
3	(6, 4)	(4, 1)	(6, 4)																																																																																																										
			(4, 1)																																																																																																										
h. GAME	i. Duopolistic Competition																																																																																																												
<table border="1" style="border-collapse: collapse; width: 100%;"> <thead> <tr> <th></th><th>1</th><th>2</th><th>3</th><th>4</th><th></th></tr> <tr> <th>1</th><td>(1, 1)</td><td>(-1, 2)</td><td>(5, 0)</td><td>(1, 1)</td><td>(0, 0)</td></tr> <tr> <th>2</th><td>(2, 3)</td><td>(1, 2)</td><td>(3, 0)</td><td>(5, 1)</td><td>(0, 9)</td></tr> <tr> <th>3</th><td>(1, 1)</td><td>(0, 5)</td><td>(1, 7)</td><td>(0, 1)</td><td>(0, 14)</td></tr> </thead> <tbody> <tr> <td></td><td></td><td></td><td></td><td></td><td>(0, 15)</td></tr> <tr> <td></td><td></td><td></td><td></td><td></td><td>(0, 12)</td></tr> <tr> <td></td><td></td><td></td><td></td><td></td><td>(0, 5)</td></tr> </tbody> </table>		1	2	3	4		1	(1, 1)	(-1, 2)	(5, 0)	(1, 1)	(0, 0)	2	(2, 3)	(1, 2)	(3, 0)	(5, 1)	(0, 9)	3	(1, 1)	(0, 5)	(1, 7)	(0, 1)	(0, 14)						(0, 15)						(0, 12)						(0, 5)	<table border="1" style="border-collapse: collapse; width: 100%;"> <thead> <tr> <th></th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th></th></tr> <tr> <th>1</th><td>(0, 0)</td><td>(0, 9)</td><td>(0, 14)</td><td>(0, 15)</td><td>(0, 12)</td><td>(0, 5)</td><td>(0, 0)</td></tr> <tr> <th>2</th><td>(9, 0)</td><td>(7, 7)</td><td>(5, 10)</td><td>(3, 9)</td><td>(1, 4)</td><td>(-1, -5)</td><td>(9, 0)</td></tr> <tr> <th>3</th><td>(14, 0)</td><td>(10, 5)</td><td>(6, 6)</td><td>(2, 3)</td><td>(-2, -4)</td><td>(-2, -5)</td><td>(14, 0)</td></tr> <tr> <th>4</th><td>(15, 0)</td><td>(9, 3)</td><td>(3, 2)</td><td>(-3, -3)</td><td>(-3, -4)</td><td>(-3, -5)</td><td>(15, 0)</td></tr> <tr> <th>5</th><td>(12, 0)</td><td>(4, 1)</td><td>(-4, -2)</td><td>(-4, -3)</td><td>(-4, -4)</td><td>(-4, -5)</td><td>(12, 0)</td></tr> <tr> <th>6</th><td>(5, 0)</td><td>(-5, -1)</td><td>(-5, -2)</td><td>(-5, -3)</td><td>(-5, -4)</td><td>(-5, -5)</td><td>(5, 0)</td></tr> </thead> <tbody> <tr> <td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>(-5, -5)</td></tr> </tbody> </table>		1	2	3	4	5	6		1	(0, 0)	(0, 9)	(0, 14)	(0, 15)	(0, 12)	(0, 5)	(0, 0)	2	(9, 0)	(7, 7)	(5, 10)	(3, 9)	(1, 4)	(-1, -5)	(9, 0)	3	(14, 0)	(10, 5)	(6, 6)	(2, 3)	(-2, -4)	(-2, -5)	(14, 0)	4	(15, 0)	(9, 3)	(3, 2)	(-3, -3)	(-3, -4)	(-3, -5)	(15, 0)	5	(12, 0)	(4, 1)	(-4, -2)	(-4, -3)	(-4, -4)	(-4, -5)	(12, 0)	6	(5, 0)	(-5, -1)	(-5, -2)	(-5, -3)	(-5, -4)	(-5, -5)	(5, 0)								(-5, -5)		
	1	2	3	4																																																																																																									
1	(1, 1)	(-1, 2)	(5, 0)	(1, 1)	(0, 0)																																																																																																								
2	(2, 3)	(1, 2)	(3, 0)	(5, 1)	(0, 9)																																																																																																								
3	(1, 1)	(0, 5)	(1, 7)	(0, 1)	(0, 14)																																																																																																								
					(0, 15)																																																																																																								
					(0, 12)																																																																																																								
					(0, 5)																																																																																																								
	1	2	3	4	5	6																																																																																																							
1	(0, 0)	(0, 9)	(0, 14)	(0, 15)	(0, 12)	(0, 5)	(0, 0)																																																																																																						
2	(9, 0)	(7, 7)	(5, 10)	(3, 9)	(1, 4)	(-1, -5)	(9, 0)																																																																																																						
3	(14, 0)	(10, 5)	(6, 6)	(2, 3)	(-2, -4)	(-2, -5)	(14, 0)																																																																																																						
4	(15, 0)	(9, 3)	(3, 2)	(-3, -3)	(-3, -4)	(-3, -5)	(15, 0)																																																																																																						
5	(12, 0)	(4, 1)	(-4, -2)	(-4, -3)	(-4, -4)	(-4, -5)	(12, 0)																																																																																																						
6	(5, 0)	(-5, -1)	(-5, -2)	(-5, -3)	(-5, -4)	(-5, -5)	(5, 0)																																																																																																						
							(-5, -5)																																																																																																						

Figure 6: Detailed payoff matrices used in matrix game environments.

be selected at each turn. These explicit legality requirements provide a clear supervision signal for action validity, which further strengthens the model’s instruction-following and rule-compliance capabilities. Prior cognitive and educational studies have also shown that turn-based board games like *TicTacToe* are effective for training planning and strategic reasoning skills (Noda et al., 2019), supporting its suitability as an informal learning environment.

To enhance training robustness and reduce prompt-specific bias, we design four distinct initial prompts that describe the game setting and interaction protocol from different perspectives. Moreover, we observe that the win conditions are not always trivially recognized by LLMs, especially in multi-turn settings. To address this issue, we further introduce an auxiliary win-condition prompt that explicitly explains the winning criteria, including illustrative examples such as horizontal, vertical, and diagonal line completions. During training, the initial prompt is randomly sampled from the prompt set, and the win-condition prompt is also randomly included to improve generalization. Qwen3-14B is employed as the opponent model during training, considering its strong overall ca-

pabilities and efficient local inference speed. An illustrative example of the prompt template with the win-condition description is provided below:

```

##Game Rules: TicTacToe
**Objective**: Be the first player to
connect 3 of your pieces in a
continuous line.
**Player Pieces**:
- Player 1: 'O'
- Player 2: 'X'
- Empty Slot: '.'
**How to Play**:
1. The game is played on a 3x3
vertical grid.
2. Players take turns setting one of
their pieces into any available
slot.
**Winning Conditions**:
The game ends when a player forms a
line of 3 of their own pieces. The
line can be:
1. **Horizontal** (side-by-side in a
row)
*Example of a horizontal win for
Player 1 ('O')*:
...
X . .
0 0 0 <-- 3 'O's in row 2
. X .
...
2. **Vertical** (stacked on top of

```

```

    each other in a column)
*Example of a vertical win for Player
  2 ('X'):  

```
.

. X 0

0 X 0 <-- 3 'X's in column 2

.: .

3. **Diagonal** (connected at an
 angle)
*Example of a diagonal win (bottom-
 left to top-right) for Player 1:

```
.. 0  

. 0 X <-- 3 '0's in a diagonal line  

0 X .  

```
*Example of another diagonal win (top
 -left to bottom-right) for Player
 2:

```
X . 0  

. X 0 <-- 3 'X's in a diagonal line  

. 0 X  

```
Draw Condition:
If the entire grid is filled with
 pieces and no player has won, the
game is a draw.

Current Game State
{state_prompt}

Your Turn
You are {player}.
The available actions are: {actions}.

```

### A.3 Who's the Spy

*Who's the Spy* is a multi-player, multi-turn social deduction game involving  $N$  players, among whom  $N - 1$  are assigned as *Civilians* and one as the *Undercover*. All players are secretly assigned a word: the civilians share the same word, while the undercover receives a different but semantically related word. Importantly, players do not know anyone's identity throughout the game.

The game proceeds in multiple speaking rounds. In each round, players take turns generating short descriptions of their assigned word, aiming to convey its meaning without revealing too much explicit information. After two full speaking rounds, all players simultaneously vote to eliminate one suspected undercover. The player receiving the most votes is removed from the game. The civilians win if the undercover is successfully eliminated, while the undercover wins if at least one civilian is voted out.

Winning *Who's the Spy* requires a delicate balance between informativeness and concealment. Civilian players must describe the shared word

accurately enough to signal alignment with other civilians, while avoiding overly explicit descriptions that could allow the undercover to infer the civilians' word and adapt accordingly. Conversely, the undercover must generate plausible but strategically ambiguous descriptions to blend in and avoid detection. This interaction demands players to reason about others' beliefs, intentions, and linguistic strategies, thereby strongly engaging theory-of-mind capabilities. At the same time, the open-ended nature of word description encourages flexible and creative language generation.

In our experiments, we fix the number of players to four, consisting of three civilians and one undercover, and set the Qwen3-14B model as training opponents. For each game instance, the identity and speaking order of the trained LLM are randomly assigned. A trajectory is considered successful if the side corresponding to the identity assigned to the trained model wins the game. The word list used in our experiments is adopted from an open-source resource<sup>2</sup>. To improve training robustness and reduce sensitivity to prompt phrasing, we design four distinct rule prompts that describe the game mechanics and player objectives from different perspectives. During training, one rule prompt is randomly sampled for each episode. An example of such a rule prompt is shown below:

```

Game: Who's the Undercover Agent
Roles:
- 3 Civilians share one word.
- 1 Undercover has a related but
different word.

Goal:
- Civilians: Find the undercover.
- Undercover: Stay hidden until only
 2 players remain.

How to Play:
Each player describes their word in
one sentence (without saying the
word itself).
Be subtle yet clear. After two-turn
speak, everyone votes out one
player. The one with most votes is
eliminated.

Win:
- Civilians win if the undercover is
 voted out.
- Undercover wins if one civilian is
 voted out.

```

<sup>2</sup>[https://github.com/xzx34/SocialMaze/tree/main/find\\_the\\_spy](https://github.com/xzx34/SocialMaze/tree/main/find_the_spy)

During the testing phase, we observe that the model tends to repeatedly generate highly similar or identical descriptions across turns, which reduces linguistic diversity and weakens the effectiveness of social deduction training. To mitigate this issue, we introduce an additional hint prompt during the description phase to encourage more varied and informative language generation:

```
Additional Rules for Description
(Very Important)
- Your description MUST be clearly
 different from any descriptions
 you have given in earlier rounds.
- Do NOT reuse similar words,
 sentence structures, or ideas.
 Avoid describing it as a "process"
 again.
- Each round, pick a NEW angle of
 interpretation (e.g., its effect,
 its form, its symbolism, its usage
 , etc.)
- The new description should have LOW
 semantic similarity with your
 previous descriptions.
- Act as if you don't remember your
 previous answers, but you MUST
 ensure this answer is not similar
 to them.
- Always produce a new, creative, and
 distinct sentence.
- If the word has multiple meanings,
 assume 100% that the basic
 meaning is intended. Never choose
 the less common or technical ones.
```

## B Detailed Training settings

### B.1 Optimization Objective

We adopt a trajectory-based reinforcement learning formulation using Group Relative Policy Optimization (GRPO) as the optimization backbone following the RAGEN framework(Wang et al., 2025c). In multi-turn settings, the model interacts with the environment over a full trajectory  $\tau_i$ , and a scalar reward is assigned at the trajectory level. The resulting advantage is normalized and distributed across all token positions within the trajectory.

Given a group of  $G$  sampled trajectories  $\{\tau_i\}_{i=1}^G$ , we define the policy ratio at token position  $t$  as

$$r_{i,t}(\theta) = \frac{\pi_\theta(\tau_{i,(t)} \mid \tau_{i,< t})}{\pi_{\text{old}}(\tau_{i,(t)} \mid \tau_{i,< t})}, \quad (3)$$

and its clipped version as

$$\tilde{r}_{i,t}(\theta) = \text{clip}(r_{i,t}(\theta), 1 - \epsilon, 1 + \epsilon). \quad (4)$$

The trajectory-level GRPO objective is then given by

$$\frac{1}{G} \sum_{i=1}^G \frac{1}{|\tau_i|} \sum_{t=1}^{|\tau_i|} \min \left[ r_{i,t}(\theta) \hat{A}_{i,t}, \tilde{r}_{i,t}(\theta) \hat{A}_{i,t} \right], \quad (5)$$

where  $\hat{A}_{i,t}$  denotes the normalized advantage at token position  $t$  within trajectory  $\tau_i$ , and  $\epsilon$  is the clipping threshold.

This objective follows a standard PPO-style formulation and serves as a unified optimization backbone for all training settings in this work.

### B.2 Nested Training Framework

Nested training constructs a composite learning task by sequentially composing multiple sub-tasks into a single trajectory. Specifically, a full trajectory in nested training consists of the concatenation of trajectories from all constituent sub-tasks, executed in sequence within one episode. Each sub-task preserves its original interaction protocol and success condition, while the model is required to solve all sub-tasks within a unified rollout.

The reward for a nested trajectory is defined as the average success across all sub-tasks, rather than a strict conjunction (i.e., requiring all sub-tasks to succeed simultaneously). This design choice avoids making the optimization problem excessively difficult, especially for 1.5B models in early training stages, while still enforcing joint task competence. Partial success on a subset of sub-tasks yields intermediate rewards, providing informative gradients that encourage balanced learning across tasks.

This formulation fundamentally differs from naive task mixing. In mixed training, each trajectory corresponds to a single sub-task sampled from a task pool, and the optimization objective aggregates rewards across trajectories from different tasks. In contrast, nested training embeds all sub-tasks within the same trajectory, forcing the model to reason across multiple task contexts in a single rollout. As a result, gradients are jointly influenced by all sub-tasks, reducing the risk that learning is dominated by easier tasks and improving credit assignment for harder ones.

From an optimization perspective, nested trajectories exhibit higher action and state diversity compared to single-task trajectories, leading to increased entropy during training. This property empirically contributes to improved gradient stability and more robust exploration. Moreover, since the

nested reward is computed as an average over sub-task rewards, the overall objective is invariant to the ordering of sub-tasks within a trajectory. We empirically verify this order-invariance property in our ablation studies, where different nesting orders yield comparable performance.

### B.3 Training Configuration and Implementation

Following the RAGEN framework based on VeRL (Sheng et al., 2024), we utilize the stable StarPO\* algorithm. The key improvements in StarPO\* are DAPO (Yu et al., 2025) and trajectory filtering. DAPO removes the KL-term and employs a clip-higher strategy. The trajectory filtering is based on variance, where only the top 25% of trajectories with the highest variance are retained for each round.

Regarding hyperparameters, each setting runs a maximum of 250 rollout-update iterations, with the group size fixed to 16. During training, we fixed the random seed to generate the task prompts to ensure reproducibility.

The maximum number of turns for each sub-task is determined by the nature of the task. For single-turn tasks such as *Math* and *Matrix Games*, the maximum number of turns is 1. For multi-turn tasks, the maximum turns for *TicTacToe* is set to 5, since the game involves a 3x3 grid, and the game will always end after the 5th turn. In *Who’s the Spy*, the maximum turns is 3, consisting of two description rounds and one voting round. In mixed training, the `max_turn` is set to the maximum number of turns across all sub-tasks. In nested training, the `max_turn` is the sum of the maximum number of turns for all sub-tasks.

The model is optimized using Generalized Advantage Estimation (GAE) with  $\gamma = 1.0$  and  $\lambda = 1.0$ , along with the Adam optimizer where  $\beta_1 = 0.9$  and  $\beta_2 = 0.999$ . We also apply entropy regularization with a coefficient  $\beta = 0.001$ . The reward is defined as 1 for success and 0 for failure, with a reward of 0.5 assigned to draws in *TicTacToe*. The format penalty of  $-0.1$  is applied. In addition, for the  $F + I_3$  setting with the 1.5B model, nested training is performed after a brief warm-up stage using a small amount of mixed-task data. This design is motivated by the observation that directly training the 1.5B model from the base initialization on the full combination of all tasks leads to unstable optimization and frequent failure to complete all sub-tasks. The warm-up phase

serves as a cold-start mitigation strategy, enabling the model to acquire basic competence across individual tasks before being exposed to the more challenging nested training objective.

In multi-task settings, both mixed and nested training scenarios involve an equal number of tasks, ensuring fairness in task comparison.

## C Detailed Evaluation Settings

We introduce the detailed evaluation settings including math, game and general abilities scenarios.

### C.1 Math Evaluation

We evaluate the model’s math reasoning ability using the MATH500 benchmark. Model outputs are compared against ground-truth answers using the Math-Verify toolkit<sup>3</sup>, which provides a robust verification pipeline for mathematical equivalence and correctness.

### C.2 Game Evaluation

To assess game-playing ability, we evaluate the model’s success rate over 100 independent game rounds, with the opponent model fixed to Gemini-2.5-Flash for *TicTacToe* and *Who’s the Spy* tasks. For each game, the success criterion is defined in accordance with the corresponding training objective. Specifically, in *Matrix Games*, a trajectory is considered successful if the selected action satisfies the Nash Equilibrium condition. In *TicTacToe*, success is defined as achieving a valid three-in-a-row configuration (horizontal, vertical, or diagonal) for the trained model’s pieces. To make the evaluation of *TicTacToe* fair, models are tested on an empty board and the first player is randomly selected. In *Who’s the Spy*, a trajectory is counted as successful if the side corresponding to the trained model’s assigned identity wins the game.

### C.3 General Abilities Evaluation

To evaluate the model’s generalization across diverse ability dimensions, we select a set of widely used benchmarks, including MMLU, MMLU-Pro, CommonGen, and SocialIQA.

**MMLU and MMLU-Pro.** MMLU covers a broad range of academic subjects and evaluates knowledge and reasoning through four-choice questions. MMLU-Pro extends MMLU by increasing both task difficulty and answer space, expanding each question from four options to ten options,

<sup>3</sup><https://github.com/huggingface/Math-Verify>

thereby posing a more challenging evaluation setting.

**CommonGen.** The CommonGen benchmark is designed to evaluate the model’s language generation ability by requiring it to produce a coherent and plausible sentence that incorporates a given set of concept words with specified semantic roles, as illustrated in Fig. 8. Since CommonGen is an open-ended generation task, we follow the evaluation protocol provided by allenai<sup>4</sup>, using few-shot prompting and GPT-4o as an automatic evaluator to compare model outputs against human-annotated reference sentences. To reduce positional bias during judgment, we randomly permute the order of the model-generated output and the reference sentence, and instruct the evaluator to carefully compare both candidates. Since the human-annotated references represent a strong upper bound that is difficult to consistently surpass, we report the success rate using a win-or-tie criterion. The detailed evaluation prompt is provided below:

```
Data
Given several concepts (i.e., nouns
or verbs), we ask models to write
a short and simple sentence that
contains *all* the required words.
The sentence should describe a common
scene in daily life, and the
concepts should be used in a
natural way.
Concepts: "{concept_list}"
Model A: "{candidate_A}"
Model B: "{candidate_B}"

Your Task
Your task is to choose a better
sentence from the two candidates.
Decide which model's sentence is
better in terms of the naturalness
and commonness of the scenes they
describe.

Rules:
- A better sentence should describe a
common scene in daily life, and
all concepts should be used in a
natural way.
- You should prefer sentences that
use all given concepts with
correct part-of-speech tags.
- A simpler and shorter sentence is
preferred if it describes the same
scene as the other sentence.
- If you think both sentences are
equally good or bad, please choose
"tie".
```

Now, please output your choice ("A" or "B" or "tie").

Your choice:

**SocialIQA.** SocialIQA is a question-answering benchmark designed to evaluate social reasoning through three-choice questions. Unlike prior benchmarks that emphasize physical or taxonomic knowledge, SocialIQA focuses on understanding people’s actions and their social motivations. Given a described action, the model must infer the most plausible social intent or implication among multiple candidates. The dataset covers a wide range of everyday social situations, with answer options consisting of both human-written and adversarially filtered machine-generated candidates.

## D Detailed Ablation Study

The detailed ablation study tables are provided here, including the opponent sensitivity analysis and the order sensitivity analysis.

**Opponent Sensitivity.** We first analyze the sensitivity of training performance to the choice of opponent model. Specifically, we compare Gemini-2.5-Flash and Qwen3-14B as fixed opponents for *TicTacToe* and *Who’s the Spy* training under the 1.5B setting, as shown in Table 5.

Overall, using Qwen3-14B as the opponent consistently leads to stronger performance across most in-domain and out-of-domain benchmarks. In particular, for *TicTacToe*, training against Qwen3-14B yields substantially higher game success rates and improved generalization on MMLU, SocialIQA, and the averaged score. Similar trends are observed for *Who’s the Spy*, where Qwen3-14B as the opponent results in competitive outcomes.

Beyond performance gains, Qwen3-14B also offers practical advantages in training efficiency. As a locally deployed model, it provides significantly faster and more stable inference compared to closed-source APIs, enabling scalable and reproducible self-play training. These results suggest that a strong yet efficient local opponent can provide higher-quality interaction signals, leading to more effective informal learning.

**Order Sensitivity.** We further investigate whether the ordering of sub-tasks in nested training affects final performance. Table 6 reports results for two different nesting orders of *Math* and *Matrix Games* using the 7B model.

<sup>4</sup><https://github.com/allenai/CommonGen-Eval>

| Setting       | Opponent         | MATH  | Matrix | TicTacToe | Spy   | MMLU  | MMLU-Pro | Common | Social | Avg.  |
|---------------|------------------|-------|--------|-----------|-------|-------|----------|--------|--------|-------|
| TicTacToe     | Gemini-2.5-Flash | 25.80 | 17.00  | 46.00     | 12.00 | 39.00 | 13.79    | 14.32  | 51.18  | 29.57 |
|               | <b>Qwen3-14B</b> | 22.40 | 32.00  | 75.00     | 21.00 | 47.89 | 16.45    | 11.56  | 64.12  | 35.00 |
| Who's the Spy | Gemini-2.5-Flash | 19.20 | 19.00  | 2.00      | 38.00 | 46.86 | 17.21    | 20.85  | 59.57  | 36.12 |
|               | <b>Qwen3-14B</b> | 19.20 | 21.00  | 0.00      | 33.00 | 43.85 | 15.68    | 20.85  | 60.70  | 35.27 |

Table 5: Opponent sensitivity analysis of *TicTacToe* and *Who's the Spy* training for base 1.5B model, comparing Gemini-2.5-Flash and Qwen3-14B as opponents.

| Model  | Order                      | MATH  | Matrix | TicTacToe | Spy   | MMLU  | MMLU-Pro | Common | Social | Avg.         |
|--------|----------------------------|-------|--------|-----------|-------|-------|----------|--------|--------|--------------|
| nested | <i>Math → Matrix Games</i> | 57.40 | 43.00  | 33.00     | 33.00 | 68.33 | 35.56    | 41.46  | 74.46  | <b>54.95</b> |
|        | <i>Matrix Games → Math</i> | 58.40 | 58.00  | 33.00     | 45.00 | 70.48 | 38.56    | 35.43  | 73.90  | <b>54.59</b> |

Table 6: Order sensitivity analysis of nested training on *Math* and *Matrix Games* using the 7B model.

While slight differences can be observed across individual benchmarks depending on the task order, the overall performance remains highly consistent. In particular, the averaged scores across all evaluation metrics are nearly identical for different orders, indicating that nested training is insensitive to the specific sequencing of sub-tasks. This empirical robustness aligns with the design of the nested objective, where rewards are computed as an average over sub-task successes, making the optimization objective invariant to task order.

Together, these ablation results demonstrate that the proposed framework is robust to both opponent selection and task ordering, further supporting the stability and general applicability of nested training.

## E Detailed Case Studies

**Effectiveness of Informal Learning on Mathematical Reasoning.** To provide a more detailed illustration of how informal learning improves mathematical reasoning, we present a representative case study on the MATH500 benchmark in Fig. 7. The example compares the base *Qwen2.5-7B-Instruct* model with the same model after training on *Matrix Games*.

In this example, both models correctly identify that if a quadratic polynomial with integer coefficients has  $4 - \sqrt{11}$  as a root, then its conjugate  $4 + \sqrt{11}$  must also be a root. The base model further recognizes the high-level structure of the solution and attempts to construct the corresponding polynomial. However, it commits a subtle arithmetic error when expanding  $(x - 4)^2 - 11$ , incorrectly computing  $16 - 11$  as 3. This local mistake propagates to the subsequent evaluation of  $p(3)$  and  $p(4)$ , ultimately leading to an incorrect final answer, despite the overall reasoning strategy being correct.

In contrast, the model trained with *Matrix Games* exhibits a more structured and explicit derivation. Instead of directly simplifying intermediate expressions, it expands the polynomial step by step as

$$(x - 4)^2 - 11 = x^2 - 8x + 16 - 11,$$

making the critical arithmetic operation transparent and easy to verify. Moreover, it explicitly observes that the leading coefficient  $a$  can be any integer and chooses  $a = 1$  for simplicity, demonstrating greater flexibility and creativity in solution construction. This explicit and verifiable reasoning process helps the model avoid local arithmetic errors and arrive at the correct final result. This case suggests that informal learning through interactive game environments encourages more cautious, structured, and self-checking reasoning behaviors in mathematical problem solving.

**Effectiveness of Nested Training on Semantic Generation.** We further analyze the difference between mixed and nested training strategies through a detailed case study on the CommonGen benchmark, as shown in Fig. 8. The task requires generating a coherent daily-life sentence that naturally incorporates all given concepts.

The model trained with mixed objectives is able to correctly include the required concepts and produce a grammatically valid sentence. However, its generation primarily focuses on satisfying individual lexical constraints, resulting in a relatively minimal and less vivid description. While the output is semantically acceptable, it lacks explicit scene construction and deeper integration among the concepts.

Question: Let  $p(x)$  be a quadratic polynomial with integer coefficients which has  $4 - \sqrt{11}$  as a root. Compute  $\frac{p(3)}{p(4)}$ .

#### Base Model: Qwen2.5-7B-Instruct

<think> Since  $p(x)$  is a quadratic polynomial with integer coefficients and  $4 - \sqrt{11}$  is a root, its conjugate  $4 + \sqrt{11}$  must also be a root. This is because the coefficients are integers, and the polynomial must factor over the reals to maintain integer coefficients. The polynomial can thus be written as  $p(x) = a(x - (4 - \sqrt{11}))(x - (4 + \sqrt{11})) = a((x-4)^2 - 11) = a(x^2 - 8x + 3)$ . We need to find  $\frac{p(3)}{p(4)}$ . First, calculate  $p(3)$  and  $p(4)$ .  
 For  $p(3)$ :  $p(3) = a(3^2 - 8 \cdot 3 + 3) = a(9 - 24 + 3) = a(-12)$   
 For  $p(4)$ :  $p(4) = a(4^2 - 8 \cdot 4 + 3) = a(16 - 32 + 3) = a(-13)$   
 Thus,  $\frac{p(3)}{p(4)} = \frac{-12}{-13} = \frac{12}{13}$ . The value of  $a$  cancels out, so the result is independent of  $a$ .</think>

<answer>  $\frac{12}{13}$ </answer>

#### Model Trained on Matrix Games

<think> Since  $p(x)$  is a quadratic polynomial with integer coefficients and  $4 - \sqrt{11}$  is a root, its conjugate  $4 + \sqrt{11}$  must also be a root. This is because the coefficients are integers, and the polynomial must have roots that come in conjugate pairs if they are irrational.  
 Let  $p(x) = a(x - (4 - \sqrt{11}))(x - (4 + \sqrt{11}))$ . Expanding this, we get:  
 $p(x) = a(x - 4 + \sqrt{11})(x - 4 - \sqrt{11}) = a((x - 4)^2 - (\sqrt{11})^2) = a(x^2 - 8x + 16 - 11) = a(x^2 - 8x + 5)$ .</think>  
 Since the leading coefficient  $a$  can be any integer, we can choose  $a = 1$  for simplicity, giving us:  
 $p(x) = x^2 - 8x + 5$ .  
 Now, we need to compute  $\frac{p(3)}{p(4)}$ :  
 $p(3) = 3^2 - 8 \cdot 3 + 5 = 9 - 24 + 5 = -10$ ,  
 $p(4) = 4^2 - 8 \cdot 4 + 5 = 16 - 32 + 5 = -11$ .  
 Thus,  $\frac{p(3)}{p(4)} = \frac{-10}{-11} = \frac{10}{11}$ . The final answer is  $\frac{10}{11}$ .</think>

<answer>  $\frac{10}{11}$ </answer>

Figure 7: An example of the effectiveness of informal learning on the MATH500 benchmark, with Qwen2.5-7B-Instruct base model trained on *Matrix Games*.

```
Instruction
Given several concepts (i.e., nouns or verbs), write a short and simple sentence that contains *all* the required words.
The sentence should describe a common scene in daily life, and the concepts should be used in a natural way.
Examples
Example 1
- Concepts: "dog(noun), frisbee(noun), catch(verb), throw(verb)"
- Sentence: The dog catches the frisbee when the boy throws it into the air.
Example 2
- Concepts: "apple(noun), place(verb), tree(noun), pick(verb)"
- Sentence: A girl picks some apples from a tree and places them into her basket.
Your Task
- Concepts: "food(noun), front(noun), sit(verb), table(noun)"
```

#### 7B Model after Mixed Training

<think> The sentence needs to include all the given concepts in a natural daily life scenario. "Food" can be something placed on a table, "front" can refer to the front of the table, and "sit" can describe the action of someone sitting down.</think>

<answer> Sara sits at the table in front of her plate of food.</answer>

#### 7B Model after Nested Training

<think> The sentence needs to include "food," "front," "sit," and "table." A common scene could be people sitting at a table and placing food in front of each other.</think>

<answer> They sit at the table and place food in front of each other for dinner.</answer>

**Ground Truth:** The boy sat at the front of the table and waited for his food.

Figure 8: An example of the effectiveness of nested training on the CommonGen benchmark, comparing mixed training with nested training on F + I<sub>1</sub> settings.

In contrast, the model trained with nested objectives demonstrates a more deliberate generation process. Before producing the final sentence, it briefly verifies the required concepts and then constructs a coherent common scenario that naturally binds them together, such as people sitting at a table and placing food in front of each other. This leads to a more expressive, contextually rich, and human-aligned sentence. As a result, the nested-trained model achieves a win-or-tie outcome when compared against strong human-annotated references.

This comparison highlights that nested training encourages deeper semantic integration and scene-level reasoning, going beyond word-level

constraint satisfaction. By requiring the model to jointly consider multiple objectives within a single trajectory, nested training promotes more coherent and expressive language generation than naive mixed training.

## F Additional Results of Closed-Source models

We further evaluate a set of strong closed-source models, including GPT-4o, Gemini-3-Flash-Preview, and DeepSeek-V3.2 (chat mode, without reasoning) (Liu et al., 2025a). All evaluation protocols are strictly aligned with those used in the main experiments. The corresponding results are reported in Table 7.

| Model         | MATH  | Matrix | TicTacToe | Spy   | MMLU  | MMLU-Pro | Common | Social | Avg.  |
|---------------|-------|--------|-----------|-------|-------|----------|--------|--------|-------|
| GPT-4o        | 59.80 | 48.00  | 63.00     | 70.00 | 85.08 | 66.00    | 53.02  | 79.02  | 70.78 |
| Gemini3-Flash | 49.80 | 64.00  | 86.00     | 16.00 | 78.42 | 52.01    | 41.96  | 74.87  | 61.82 |
| DeepSeek-V3.2 | 55.80 | 61.00  | 88.00     | 69.00 | 86.73 | 63.69    | 67.34  | 80.19  | 74.49 |

Table 7: Additional results of close-source models.

Overall, these results demonstrate that strong closed-source models exhibit competitive performance across both formal and informal learning benchmarks, while showing distinct strengths on different task categories. GPT-4o achieves balanced performance across mathematical reasoning, social reasoning, and general knowledge benchmarks, reflecting its strong general-purpose capability. Gemini-3-Flash-Preview performs particularly well on interactive game tasks such as *TicTacToe* and *Matrix Games*, but shows relatively weaker performance on social deduction and general language generation tasks. DeepSeek-V3.2 achieves the highest overall average score, with consistently strong results on math, game, and general ability benchmarks, indicating that competitive generalization can be achieved even without explicit reasoning traces. These observations suggest that while closed-source models benefit from large-scale training and strong base capabilities, their performance profiles vary substantially across different ability dimensions.

Moreover, we observe that the 1.5B and 7B models after RL training can, under certain domain-specific or nested training settings, achieve performance comparable to or even surpass strong closed-source models on targeted benchmarks. As shown in the main results, a 1.5B model trained under the F + I<sub>1</sub> setting attains 65.00% accuracy on *Matrix Games*, exceeding the performance of all evaluated closed-source models in this domain.

Similar phenomena are also observed for 7B models on several general ability benchmarks, including MMLU-Pro, CommonGen, and SocialIQA, where models trained with nested or domain-aligned tasks achieve results on par with those of closed-source APIs. These gains are particularly pronounced on tasks that require structured reasoning, interaction, or social inference, which are directly targeted by the corresponding informal learning environments.

Taken together, these results indicate that, despite the substantially smaller model size and more limited training data, our proposed training frame-

work can effectively elicit strong domain-specific and transferable capabilities through structured and nested reinforcement learning. This comparison with closed-source models provides further evidence that carefully designed interactive training objectives can partially compensate for scale and serve as an efficient alternative for capability enhancement.