STARTING NEAR THE GOAL: CURRICULUM REINFORCEMENT LEARNING FOR TRAINING INITIALLY WEAK LLM POLICIES

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement learning with verifiable rewards is difficult for initially weak language-model policies. An initially weak policy may rarely discover successful trajectories, and terminal rewards provide little direct guidance for intermediate reasoning decisions. To address these challenges, we propose Curriculum Reinforcement Learning (CRL), a training framework that begins exploration near the goal state and uses outcome feedback to adjust curriculum. Starting with shorter continuations reduces the difficulty of obtaining successful outcomes early in training, helping to alleviate the RL cold-start problem. Adaptive curriculum strength and infix reinforcement then support policy improvement by adjusting external guidance and reinforcing reasoning patterns from the policy's own successful experience. Together, these mechanisms make curriculum responsive to the needs of RL and facilitate the transition to autonomous reasoning as reference guidance is gradually withdrawn. Extensive evaluations across multiple mathematical reasoning benchmarks show that CRL improves both reasoning performance and training efficiency. The code is available at: https://anonymous.4open.science/r/LLM-CRL

1 INTRODUCTION

Reinforcement learning (RL) can improve the reasoning capabilities of large language models (LLMs) through outcome-level rewards (Cheng et al., 2025; Zheng et al., 2025; Wang et al., 2025). Early in training, finding a correct solution to a difficult reasoning problem can resemble solving a scrambled Rubik's Cube through trial and error (Agostinelli et al., 2019): many attempts fail before a successful sequence of decisions is discovered. When rewards depend on a successful reasoning trajectory, an initially weak policy may rarely generate responses that receive non-zero rewards (Feng et al., 2025). Consequently, the model has few successful trajectories to learn from and limited feedback to guide policy updates. This scarcity of rewards can slow policy improvement, creating a cold-start challenge for RL training. The effectiveness of policy-gradient RL for LLM reasoning is closely tied to the initial policy's ability to generate successful trajectories (Chen et al., 2026). Empirical analyses suggest that Reinforcement Learning with Verifiable Rewards gains can largely reflect more efficient sampling of successful responses already accessible to the base model (Yue et al., 2025). When such responses are difficult to obtain, sparse outcome rewards provide limited guidance for policy improvement.

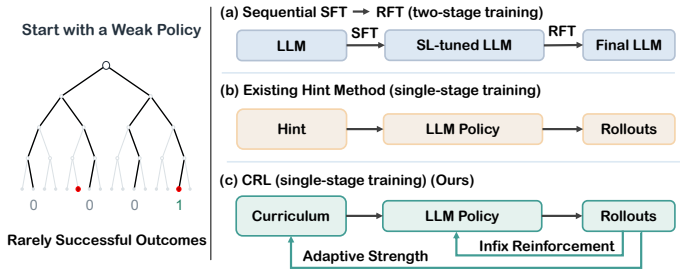

Figure 1: **Left:** An initially weak policy struggles to discover successful trajectories, resulting in sparse non-zero rewards. **Right:** (a) Sequential SFT–RFT uses a separate supervised stage to initialize RL. (b) Existing hint-based methods guide exploration during RL training. (c) CRL uses outcome feedback to adjust curriculum strength and reinforces reasoning patterns from successful continuations through infix reinforcement.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

These difficulties become more pronounced in long-horizon reasoning, where an LLM must generate many interdependent steps before receiving outcome feedback (Liu et al., 2026; Yan et al., 2025; Huang et al., 2025). Each step shapes the context for subsequent generation, so an early error can lead later reasoning away from a correct solution. As the reasoning horizon grows, the policy must explore a larger space of possible continuations, making successful trajectories harder to discover (Lu et al., 2026). Moreover, a terminal reward evaluates the response as a whole and provides little direct information about which intermediate decisions contributed to the outcome. The combination of infrequent success and average maximum likelihood estimation makes it difficult to learn effective reasoning behavior from trajectory-level rewards alone (Hao et al., 2026; Oh et al., 2018; Zelikman et al., 2022; Liu et al., 2026).

This motivates training strategies that make successful experience easier to obtain and more informative for subsequent policy updates. A common approach first uses supervised fine-tuning (SFT) to obtain a stronger initial policy (Shao et al., 2024; Guo et al., 2025). Dense token-level supervision from reference solutions helps the model learn useful reasoning patterns and makes successful responses easier to generate when RL begins. Reinforcement fine-tuning (RFT) then uses outcome rewards to further improve task performance and generalization (Guo et al., 2025). However, this sequential approach requires a separate supervised training stage, which can add training overhead.

We draw inspiration from a successful RL approach to solving the Rubik's Cube (Agostinelli et al., 2019). It generates training states by scrambling the solved cube and learns a value function whose estimates improve first for states near the goal and then for more distant states. This progression can be viewed as an implicit curriculum that makes learning possible despite sparse successful outcomes. This principle can be adapted to complex LLM reasoning by using partial reference solutions as intermediate starting states. Training can begin near the end of a reference solution and progressively move toward its beginning (Resnick et al., 2022; Liu et al., 2026), increasing the portion the policy must generate independently. However, applying this curriculum to LLMs involves more than changing the starting point (Chen et al., 2025). Success with a long curriculum does not establish that the policy can solve the problem independently, so the amount of supplied context must be considered when adjusting curriculum. Moreover, the first generated steps after the curriculum shape subsequent reasoning, and terminal rewards provide little direct feedback for improving these decisions (Xi et al., 2024). These challenges motivate a training approach that adapts curriculum to the policy's progress and reinforces reasoning patterns from its own successful continuations, supporting the transition to autonomous reasoning.

To address these challenges, we propose Curriculum Reinforcement Learning (CRL), which begins RL training from intermediate reasoning states near the end of a reference solution. Figure 1 illustrates the exploration challenge faced by an initially weak policy and compares existing training approaches with CRL. CRL partitions each response into a curriculum prefix, an initial policy-generated segment, and the remaining continuation, allowing outcome feedback to determine the learning signals applied to different token segments. A long curriculum prefix initially leaves only a short continuation for the policy to generate, making successful outcomes easier to obtain. As training progresses, the prefix is gradually shortened, moving the rollout starting point toward the beginning of the solution and extending the reasoning horizon handled by RL. Rollout success and the supplied prefix fraction jointly adjust prefix curriculum. Successful initial segments provide targets for infix reinforcement, complementing RL updates over the complete continuation. The reference prefix is eventually removed, allowing the policy to explore and generate full solutions from the problem alone. Our contributions are summarized as follows:

1. We propose CRL, a method for training initially weak LLM policies under sparse outcome rewards. Building on prefix curricula, it combines progressively longer policy-generated continuations with outcome feedback updates on curriculum and infixes.

2. We introduce an adaptive curriculum strength mechanism that jointly considers rollout success and the gradual shortening of prefixes. This feedback aligns curriculum with the needs of RL training supporting a more effective transition to autonomous RL.

3. We introduce infix reinforcement to reinforce reasoning patterns from the policy's own successful continuations. It focuses on the infix immediately following the reference prefix, encouraging the policy to retain and reuse these patterns during subsequent RL exploration.

2 RELATED WORK

Reinforcement Learning for LLM Reasoning. Reinforcement learning with verifiable rewards (RLVA) optimizes reasoning policies using task outcomes and has played an important role in recent advances in LLM reasoning. GRPO is introduced by DeepSeekMath (Shao et al., 2024), which estimates advantages from groups of sampled responses without training a separate value model. DeepSeek-R1-Zero demonstrated substantial reasoning improvements through RL directly from a pretrained model (Guo et al., 2025). However, the full DeepSeek-R1 pipeline additionally incorporates cold-start supervision and multiple training stages. SimpleRL-Zoo further showed that the effectiveness of RL depends on the initial model, training-task difficulty, and reward design (Zeng et al., 2025). When a weak initial policy rarely generates successful responses, outcome rewards provide limited guidance for improvement. ProRL reported broader problem coverage on several tasks after prolonged training with mechanisms that preserve exploration (Liu et al., 2025). At the same time, Yue et al. (2025) found that, in the RLVR settings they evaluated, improvements largely reflected more effective sampling of successful responses already accessible to the base model, highlighting the importance of the initial policy. These studies emphasize the role of initial policy capabilities and effective exploration, motivating our focus on making successful experience more accessible early in RL training.

Curriculum Learning from Intermediate States. Beyond LLM reasoning, weak initial policies also face exploration difficulties in sparse reward tasks, where successful outcomes are rarely encountered. Numerous studies address this challenge by learning from states closer to the final goal (Florensa et al., 2017; Resnick et al., 2022; Agostinelli et al., 2019). Reverse Curriculum Generation samples feasible starting states near a known goal and expands the training region according to policy performance (Florensa et al., 2017). DeepCubeA generates training states through reverse moves from the solved cube and learns a cost-to-go function through approximate value iteration to guide search (Agostinelli et al., 2019). These approaches illustrate how learning from states near the goal can make useful learning signals more accessible early in training. In LLM RL training, R^3, UFT, and Prefix-RFT use reference solutions to initialize policy generation from intermediate reasoning states (Xi et al., 2024; Liu et al., 2026; Huang et al., 2025). The supplied solutions allow the policy to begin exploration closer to the end of a solution, reducing the amount of reasoning it must generate before receiving outcome feedback. Their results demonstrate the benefits of learning from intermediate reasoning states for LLM reinforcement learning.

3 CURRICULUM REINFORCEMENT LEARNING

3.1 PRELIMINARIES

Notation. Let $\mathbf{X}$ denote an input problem and $\mathbf{Y}^\star = (y_1^\star, \ldots, y_{T^\star}^\star)$ its expert response, where $(\mathbf{X}, \mathbf{Y}^\star) \sim \mathcal{D}_{\mathrm{SL}}$. We use $\pi_{\boldsymbol{\theta}}$, π_{old}, and π_{ref} to denote the policy being optimized, the rollout policy, and the reference policy, respectively. For a response $\mathbf{Y} = (y_1, \ldots, y_{|\mathbf{Y}|})$, $\mathbf{Y}_{<t}$ denotes the prefix preceding token y_t. For each input, the rollout policy generates G responses $\{\mathbf{Y}^i\}_{i=1}^G$.

Reinforcement Learning with Verifiable Rewards. We use GRPO to optimize rewards provided by a task verifier (Shao et al., 2024). For each input $\mathbf{X}$, the rollout policy π_{old} samples G responses $\mathbf{Y}^i$, with rewards $r_i = \mathcal{R}(\mathbf{X}, \mathbf{Y}^i)$. GRPO estimates response-level advantages by normalizing rewards within the group and updates the policy through a clipped objective. We evaluate this objective using the empirical response-token average $\mathbb{E}_{\mathrm{resp}}$. With entropy and reference-policy regularization, the RL objective is

$$\mathcal{L}_{\mathrm{RL}}(\boldsymbol{\theta}) = \mathcal{L}_{\mathrm{GRPO}}(\boldsymbol{\theta}) - \lambda_{\mathrm{ent}}\mathcal{H}(\pi_{\boldsymbol{\theta}}) + \lambda_{\mathrm{KL}} D_{\mathrm{KL}}\left(\pi_{\boldsymbol{\theta}} \| \pi_{\mathrm{ref}}\right), \tag{1}$$

where λ_{ent} and λ_{KL} are fixed coefficients. The regularization terms are averaged over generated-token contexts.

3.2 RESPONSE-PART SEGMENTS

As illustrated in Figure 2, each complete response consists of a reference prefix, a policy-generated infix, and the remaining suffix. The reference prefix $\mathbf{H}_\ell$ provides an intermediate reasoning state $\mathbf{S}_\ell = \mathbf{X} \oplus \mathbf{H}_\ell$, from which the policy generates a continuation $\mathbf{Y}^i$. The infix $\mathbf{Y}^i_{\mathrm{inf}}$ is the initial

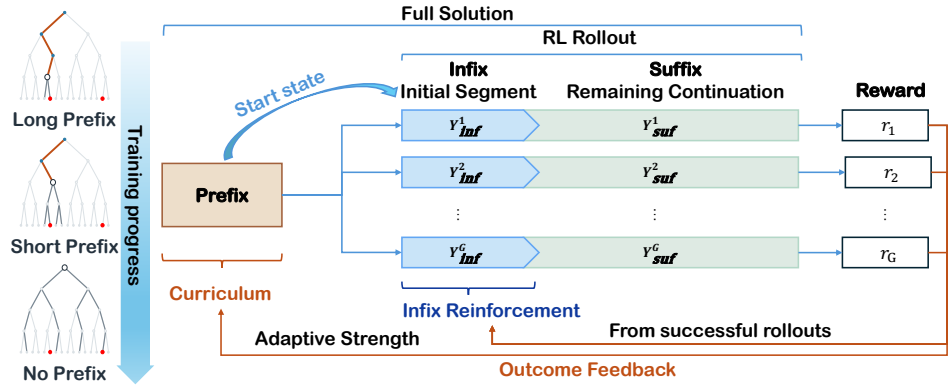

Figure 2: Overview of CRL. Reference prefixes are gradually shortened, moving rollout starting points toward the beginning of the solution and extending autonomous generation. From each shared prefix state, the policy samples G continuations, each consisting of an initial infix $\mathbf{Y}^i_{\text{inf}}$ and a remaining suffix $\mathbf{Y}^i_{\text{suf}}$. Rollout outcomes adjust curriculum strength and reinforce reasoning patterns from successful continuations through infix. Once reference guidance is fully withdrawn, both auxiliary updates are disabled and training proceeds with RL alone.

generated segment immediately following the reference prefix, with its endpoint determined by a text-boundary rule. The suffix $\mathbf{Y}^i_{\text{suf}}$ contains the rest of the continuation. The complete response is:

$$\underbrace{\mathbf{H}_\ell \,\Big|\, \mathbf{Y}^i_{\text{inf}} \,\Big|\, \mathbf{Y}^i_{\text{suf}}}_{\text{full response}}.$$

During the curriculum, the prefix provides both the starting context and a learning signal for the policy. Selected infixes from successful continuations receive additional reinforcement based on outcomes. RL updates remain active on all generated tokens, including both the infix and suffix.

3.3 Starting RL from Intermediate Reasoning States

CRL begins RL training from intermediate reasoning states near the end of a reference solution through a curriculum. A long reference prefix initially leaves only a short continuation for the policy to generate. This makes successful outcomes more accessible to an initially weak policy, providing useful experience before it can reliably generate complete solutions independently. Following UFT (Liu et al., 2026), we gradually shorten reference prefixes using an annealed sampling schedule. At training step k, the prefix cut point is sampled as:

$$\ell \sim \mathcal{P}^{\text{hint}}_k, \tag{2}$$

where $\mathcal{P}^{\text{hint}}_k$ follows a cosine annealing schedule. The schedule favors longer prefixes early in training and shorter prefixes later. As training progresses, the policy begins from earlier reasoning states and generates a larger portion of each rollout. The amount of reasoning handled by RL increases as the supplied reference context decreases. Generation always proceeds forward from the supplied prefix. Let K_{curr} denote the duration of the curriculum. After this phase, the reference prefix is removed: $\mathbf{H}_\ell = \varnothing$ and $\mathbf{S}_\ell = \mathbf{X}$. The policy then generates complete solutions from the problem alone. The curriculum thus prepares the policy for autonomous reasoning by gradually increasing the portion of the solution it must complete independently.

3.4 Outcome Feedback

As RL training progresses and prefixes are gradually shortened, the curriculum should remain responsive to the policy's learning needs. A fixed curriculum strength does not account for how much reference support the policy requires at different stages of learning. Moreover, terminal rewards provide limited direct feedback for the initial decisions made after the reference prefix. To address these challenges, we use rollout outcomes to adapt curriculum strength and reinforce reasoning patterns from successful continuations, helping the curriculum support policy improvement throughout

the transition to autonomous reasoning. Let $c_i \in 0, 1$ indicate whether continuation $\mathbf{Y}^i$ successfully solves the task. For the group generated from $\mathbf{S}_\ell$, we define $\hat{p}_\ell$ as the rollout success rate:

$$\hat{p}_\ell = \frac{1}{G} \sum_{i=1}^{G} c_i. \tag{3}$$

This success rate measures the policy's performance under the current curriculum. We also use $f_\ell \in [0, 1]$ to denote the supplied prefix fraction, measured as the proportion of reference-solution tokens included in $\mathbf{H}_\ell$.

Adaptive Curriculum Strength. A high success rate under a long reference prefix does not establish that the policy can solve the problem independently. Therefore, we use rollout success and the supplied prefix fraction together to regulate curriculum strength, defined as the weight of the reference-prefix learning signal. With a fixed base weight $\omega > 0$, we set

$$\omega_\ell^{\text{hint}} = \text{sg}\left[\omega \left(1 - (1 - f_\ell)\hat{p}_\ell\right)\right], \tag{4}$$

where sg denotes stop-gradient. When all continuations fail, the weight retains its base value. When all continuations succeed, it becomes ωf_ℓ. This preserves stronger reference support when success is achieved with substantial guidance and allows a smaller contribution when the policy succeeds with little supplied context. The weight can increase or decrease with the policy's current rollout performance.

Infix Reinforcement. The infix is the first policy-generated region after the reference prefix and serves as the transition between guided and autonomous reasoning. GRPO ordinarily applies the same trajectory-level advantage to all generated tokens, without distinguishing which intermediate decisions contributed most to the final outcome. To mitigate this coarse credit assignment, we compare infix candidates generated from the same prefix state and use their complete continuation outcomes to construct a localized learning target.

It reinforces reasoning patterns from the policy's own successful continuations by applying an additional update to selected infix tokens. This focuses the learning signal on the decisions made immediately after the reference prefix and encourages the policy to retain and reuse these patterns during subsequent exploration. Let $m_{i,t}^{\text{infix}}$ indicate whether token y_t^i belongs to the infix. With the same fixed base weight $\omega > 0$, the weight assigned to continuation i is

$$\omega_i^{\text{infix}} = \text{sg}\left[\omega \; c_i(1 - \hat{p}_\ell)\right]. \tag{5}$$

The additional outcome-conditioned update is applied only to this transition region, while the remaining suffix continues to receive the standard RL update. Successful continuations receive positive weights when the group also contains failures. The additional update is inactive when all continuations fail or all succeed. It does not claim to recover the true causal contribution of each infix decision, since its value is still estimated from the terminal outcome of the complete continuation. Rather, it localizes the additional credit to a short policy-generated region instead of applying it uniformly across the entire trajectory. The position of the infix also changes with the curriculum. Early in training, the prefix covers a large portion of the reference solution, so the infix lies near the end of the reasoning process, where completion is relatively easy. As the prefix becomes shorter, the infix moves toward earlier and more difficult reasoning states. In this sense, the model learns to complete the reasoning process from deeper states before progressively moving toward full end-to-end generation from the root problem state.

3.5 Unified Objective

We formulate a unified objective that uses curriculum guidance and outcome feedback to support RL training. We implement the two outcome-dependent updates using weighted token-likelihood losses. These losses provide additional learning signals during the curriculum. The RL objective remains active on all generated tokens throughout training.

Curriculum Objective. Let $\mathbb{E}_{\text{hint}}$ denote the unweighted empirical average over supplied prefix tokens. The prefix loss can be expressed as:

$$\mathcal{L}_{\text{hint}}(\boldsymbol{\theta}) = -\mathbb{E}_{\text{hint}}\left[\omega_\ell^{\text{hint}} \log \pi_{\boldsymbol{\theta}}(h_{\ell,t} \mid \mathbf{X}, \mathbf{H}_{\ell,<t})\right], \tag{6}$$

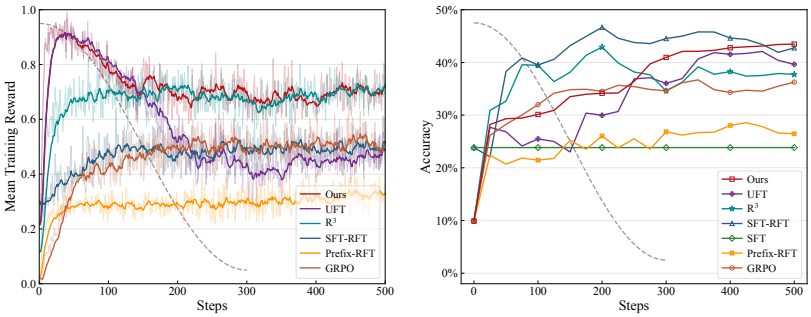

Figure 3: Training reward (left) and evaluation accuracy (right) for Qwen3.5-0.8B. The white dashed line shows how the reference prefix length changes during training.

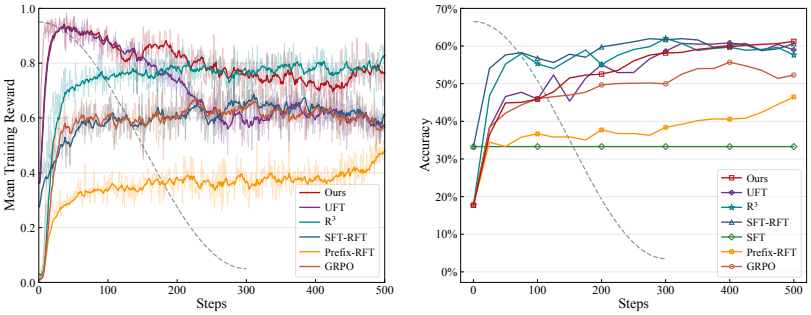

Figure 4: Training reward (left) and evaluation accuracy (right) for Qwen3.5-2B

where $h_{\ell,t}$ is the t-th token of $\mathbf{H}_\ell$. This loss helps the policy learn the reasoning supplied by the reference prefix, preparing it to generate more of that reasoning as the prefix is shortened. We define $\mathcal{L}_{\text{hint}} = 0$ when no reference-prefix tokens are supplied.

Infix Objective. The infix objective uses token-likelihood to reinforce reasoning patterns from the policy's own successful continuations. It increases the likelihood of selected infix tokens, concentrating the additional learning signal on the transition from reference guidance to autonomous generation. It can be expressed as:

$$\mathcal{L}_{\text{infix}}(\boldsymbol{\theta}) = -\mathbb{E}_{\text{resp}}\left[\omega_i^{\text{infix}} m_{i,t}^{\text{infix}} \log \pi_{\boldsymbol{\theta}}(y_t^i \mid \mathbf{S}_\ell, \mathbf{Y}_{<t}^i)\right]. \tag{7}$$

The mask restricts this additional update to infix tokens. Its normalization includes all valid generated tokens, including unsuccessful continuations and unselected positions. Both token averages retain their unweighted denominators, so the adaptive weights directly control the contributions of the corresponding losses.

Overall Objective. We set $a_k = \mathbf{1}$ when $k \le K_{\text{curr}}$; otherwise, $a_k = \mathbf{0}$. It indicates whether training is within the curriculum phase. The complete objective at step k is:

$$\mathcal{L}_{\text{RPA-RL}}^{(k)}(\boldsymbol{\theta}) = \mathcal{L}_{\text{RL}}(\boldsymbol{\theta}) + a_k\left[\mathcal{L}_{\text{hint}}(\boldsymbol{\theta}) + \mathcal{L}_{\text{infix}}(\boldsymbol{\theta})\right]. \tag{8}$$

During the curriculum, the policy generates an increasing portion of each solution, and outcome feedback adjusts the additional learning signals. Once the curriculum ends, $a_k = 0$ and $\mathbf{H}_\ell = \varnothing$. Curriculum learning and infix reinforcement are both disabled, and the objective reduces to $\mathcal{L}_{\text{RL}}$(eq. 1). The policy then generates complete solutions from the problem alone, with all subsequent updates determined by the regularized RL objective.

4 Experiments

4.1 Main Result

Overall Performance. Table 1 shows that CRL achieves the highest overall score for both Qwen3.5-0.8B and Qwen3.5-2B. Compared with GRPO, CRL improves performance across all evaluated tasks for Qwen3.5-2B and all but Countdown for Qwen3.5-0.8B. These results support the effectiveness of the proposed training framework. The improvements on the smaller model are particularly

Table 1: Math reasoning performance on in-distribution and out-of-distribution benchmarks. **Bold** denotes the best result and underline denotes the second-best result within each model group.

Method	In-Distribution		Out-of-Distribution						Avg.
	MATH	Countdown	AIME24	AMC	MATH500	Minerva	Olympiad	GSM8K	
Qwen3.5-0.8B									
Base (QwenTeam, 2026)	4.00	10.06	0.00	0.00	3.20	0.37	0.00	24.72	9.89
SFT (Ouyang et al., 2022)	17.29	37.40	3.33	4.82	21.40	7.35	2.67	35.18	23.84
RFT-only									
GRPO (Shao et al., 2024)	23.63	**72.46**	0.00	10.84	26.60	10.29	6.84	44.43	36.26
SimpleRL-Zoo (Zeng et al., 2025)	12.60	50.68	0.00	4.82	21.20	6.99	2.82	20.24	21.58
Joint SFT and RFT									
SFT-RFT (Guo et al., 2025)	29.49	65.92	0.00	15.66	38.60	13.24	10.10	**62.09**	42.76
UFT (Liu et al., 2026)	29.20	57.23	0.00	**16.87**	32.00	16.54	8.92	59.74	39.63
Prefix-RFT (Huang et al., 2025)	20.61	41.89	0.00	7.23	24.80	8.82	4.31	36.39	26.46
R^3 (Xi et al., 2024)	26.56	53.61	0.00	15.66	30.60	15.81	6.09	59.67	37.73
Ours	**34.47**	64.75	**3.33**	15.66	**41.60**	**16.91**	**11.59**	59.06	**43.47**
Qwen3.5-2B									
Base (QwenTeam, 2026)	15.82	3.22	0.00	3.61	22.20	4.41	2.08	40.71	17.71
SFT (Ouyang et al., 2022)	26.76	42.68	0.00	6.02	34.40	12.87	6.24	51.10	33.28
RFT-only									
GRPO (Shao et al., 2024)	50.88	67.38	3.33	19.28	59.00	17.65	16.05	68.01	52.30
SimpleRL-Zoo (Zeng et al., 2025)	9.96	56.84	0.00	6.02	12.80	5.15	3.86	22.74	22.19
Joint SFT and RFT									
SFT-RFT (Guo et al., 2025)	56.25	70.02	3.33	31.33	62.20	27.21	20.51	**86.66**	60.63
UFT (Liu et al., 2026)	56.25	63.48	0.00	**39.76**	64.20	26.84	**21.84**	83.78	58.98
Prefix-RFT (Huang et al., 2025)	39.45	63.96	0.00	15.66	43.60	15.44	9.66	67.55	46.46
R^3 (Xi et al., 2024)	**60.74**	64.06	6.67	**39.76**	**63.80**	**27.57**	19.02	83.09	59.51
Ours	60.55	**70.21**	**10.00**	31.33	63.60	25.74	**21.84**	84.38	**61.24**

relevant to our motivation: even when the initial policy has limited task performance, curriculum and successful rollout experience can jointly support stronger reasoning performance.

CRL also achieves higher overall scores than the sequential SFT-RFT pipeline and existing approaches that incorporate reference solutions during RL. The comparison with SFT-RFT shows that strong final performance can be achieved without a separate supervised warm-up stage. Moreover, the gains over UFT, Prefix-RFT, and R^3 are obtained using the same reference solutions. These comparisons highlight the importance of how the solution is used during policy optimization. They support the effectiveness of a training formulation that adapts curriculum using outcome feedback and reinforces selected reasoning patterns from successful continuations. The benefits also extend to autonomous reasoning on benchmarks beyond the training tasks. We achieve stronger results on Out-of-Training datasets. It supports our objective: using reference information during training to improve the policy's ability to generate complete solutions independently.

4.2 ABLATION STUDY

4.3 DYNAMIC TRAINING PROCESS

Figures 3 and 4 show the evolution of training reward and evaluation accuracy for both model sizes. CRL achieves high training rewards early in optimization, indicating that reference prefixes make successful continuations accessible before the policy can reliably generate complete solutions independently. UFT exhibits a similar initial pattern, reflecting the shared benefit of starting exploration from partial solutions. The difference becomes more apparent later in training: CRL maintains higher training rewards than UFT and GRPO as the policy generates larger portions of each solution. This suggests that its benefits extend beyond the initial guidance provided by reference prefixes.

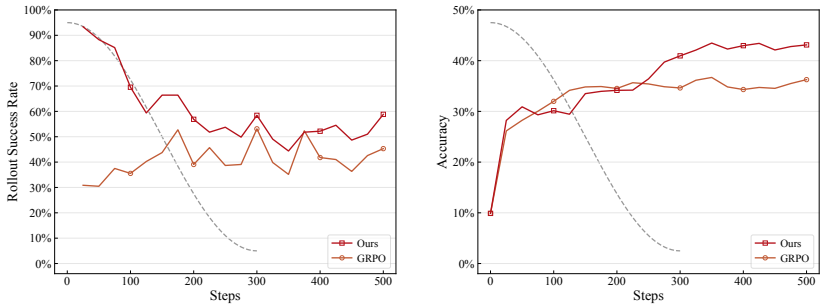

Figure 5: Qwen3.5-0.8B rollout success rate (left) and Evaluation Accuracy (right)

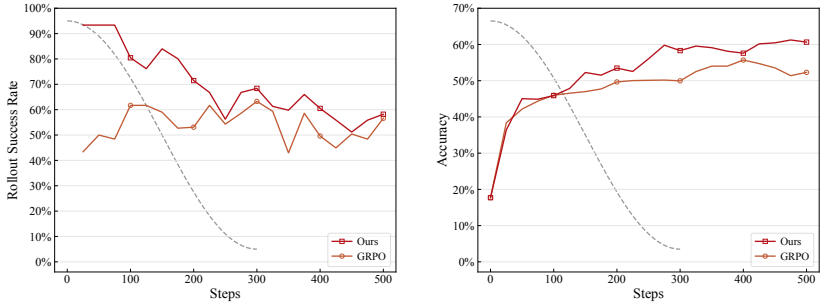

Figure 6: Qwen3.5-2B rollout success rate (left) and Evaluation Accuracy (right)

Qwen3.5-2B achieves higher early training rewards and evaluation accuracy than Qwen3.5-0.8B, supporting our motivation that stronger initial policies can more readily discover successful trajectories and thereby facilitate RL training. The evaluation curves provide complementary evidence of the transition toward autonomous RL training. Training reward declines from its early peak as reference guidance is reduced. At the same time, accuracy on evaluation without hints generally increases. These trends indicate that a decline in guided training reward does not necessarily reflect weaker reasoning performance, since the policy must complete increasingly demanding continuations during training. CRL becomes competitive with SFT-RFT at later checkpoints and retains its gains on evaluation without reference prefixes.

Table 2: Ablation on curriculum strength.

Setting	Strength	Accuracy
Fixed	0.0005	36.41
	0.001	40.63
	0.005	39.72
Adaptive	–	**43.47**

Rollout Analysis. Figures 5 and 6 show that CRL achieves higher rollout success rates than GRPO during early training. By making successful outcomes more accessible to an initially weak policy, the curriculum provides useful experience and more favorable conditions for early RL training. As the curriculum progresses, prefixes are gradually shortened and the policy generates an increasing portion of each solution. Once the curriculum ends, training will proceed with RL alone. The accuracy advantage over GRPO persists after this transition, suggesting that the successful experience obtained during the curriculum helps prepare the policy for subsequent autonomous reasoning.

Table 3: Effect of the reference solution.

Solution Source	Accuracy (%)
Qwen3.5-0.8B	32.51
Qwen3.5-27B	40.30
Ground Truth	**43.47**

Training Efficiency. Figures 7 and 8 show that CRL achieves the shortest mean rollout length and lowest mean training batch time among the compared methods for both model sizes. Starting exploration near the end of solutions allows the policy to generate shorter responses before receiving outcome feedback. This reduces token consumption during early exploration and limits the cost of collecting training rollouts. The lower batch time, which includes policy updates, further indicates that these savings translate into reduced training overhead per batch. Together with the accuracy results, these findings show that CRL improves overall reasoning performance with fewer generated tokens per rollout and less time per training batch.

Table 4: Ablation on infix reinforcement.

Infix Length	Accuracy (%)
0%	40.57
5%	41.36
10%	43.47
15%	42.34

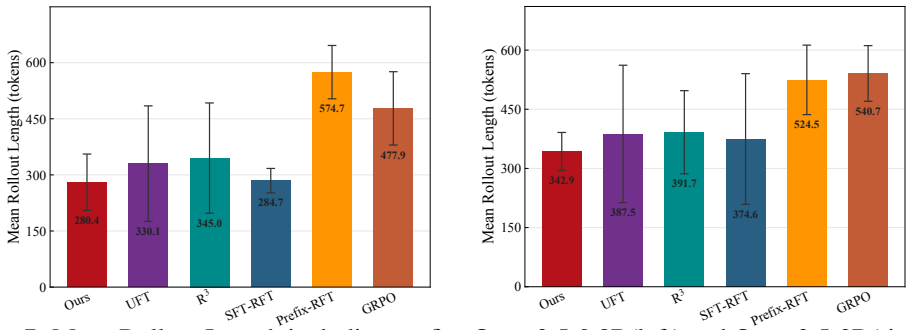

Figure 7: Mean Rollout Length including prefix: Qwen3.5-0.8B(left) and Qwen3.5-2B(right).

The Effect of Adaptive Curriculum Strength. We conduct this ablation on Qwen3.5-0.8B to examine how curriculum strength affects a weaker initial policy. Table 2 compares adaptive weighting of prefix curriculum with three fixed coefficients. The adaptive strength outperforms all tested fixed settings. Increasing the fixed weight does not consistently improve accuracy, suggesting that stronger curriculum is not uniformly more beneficial for RL. These results support adapting the influence of reference solutions to the policy's learning progress. By considering both rollout success and the supplied prefix fraction, the adaptive rule accounts for the amount of guidance under which successful generation is achieved. Its improved performance on evaluation without prefixes supports the use of adaptive curriculum to help weak initial policies develop autonomous reasoning. In addition, we evaluated the quality of the reference solutions. As shown in Table 3, using ground-truth reference solutions yields the highest accuracy. It indicates that the quality of reference solutions plays an important role in the effectiveness of CRL.

The Effect of Infix Reinforcement. Table 4 compares CRL with and without infix reinforcement. Removing this component reduces evaluation accuracy, supporting the value of infix reinforcement as an additional learning signal during RL training. The infix objective reinforces reasoning patterns from successful continuations at the point where the policy begins generating beyond the prefix. This gives the initial generated segment additional training emphasis alongside the trajectory-level RL update. The improvement on evaluation without prefixes suggests that reinforcing these patterns during guided training can benefit subsequent autonomous reasoning. Moreover, applying the infix objective to a randomly selected region can reduce accuracy below that achieved without infix reinforcement (Table 5), highlighting the importance of where this additional update is applied.

Table 5: Ablation on infix position.

Infix Position	Accuracy (%)
Random Region	39.23
Prefix-Infix-Suffix	43.47

5 Conclusion

In this paper, we introduce Curriculum Reinforcement Learning(CRL), an initially weak policy training framework that supports reinforcement learning for complex LLM reasoning under sparse non-zero rewards. CRL starts exploration near the end of reference solutions and gradually extends the portion generated by the policy, making successful experiences more accessible early in training. Outcome feedback adjusts the strength of the reference curriculum, and infix reinforcement reinforces reasoning patterns from successful continuations. CRL can achieve performance comparable to a two-stage SFT-RFT training pipeline. Experiments across two model sizes show improved overall performance over the evaluated baselines.

Limitations. Our current formulation assumes access to a curriculum reasoning solution from which the training prefix and infix granularity can be derived. Therefore, it cannot be directly applied to datasets containing only final answers without introducing an alternative source of intermediate reasoning. Besides, the feedback signal is available only when the rollout group contains sufficiently diverse candidates with distinguishable outcomes.

Future work. We will construct guidance from self-generated trajectories, incorporate more robust feedback for open-ended tasks, and develop finer-grained credit estimation beyond terminal rewards.

AI USE STATEMENT

In this work, we used generative AI tools to create and edit source code and to improve the readability and clarity of the paper. We reviewed all AI-assisted code and verified its correctness through testing, and we reviewed all AI-assisted manuscript changes for consistency and accuracy. We take responsibility for the final content of this work, including all texts and code produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The details needed to reproduce our experiments are provided in the appendix, including the training setup and hyperparameters specific to each experiment described in the main paper. All source code necessary for our proposed CRL is provided in the supplemental material via an anonymized codebase. The appendix also provides details on the prompts, datasets, and experimental parameters used in our method.

REFERENCES

Forest Agostinelli, Stephen McAleer, Alexander Shmakov, and Pierre Baldi. Solving the rubik's cube with deep reinforcement learning and search. *Nature Machine Intelligence*, 1(8):356–363, 2019.

Shuang Chen, Hangyu Guo, Zhaochen Su, Yafu Li, Jiacheng Chen, Yulun Wu, Weijie Wang, ZhiYuan Feng, Xiaoye Qu, and Yu Cheng. Revisual-r1: Advancing multimodal reasoning from optimized cold start to staged reinforcement learning. In *International Conference on Learning Representations*, volume 2026, pp. 32869–32888, 2026.

Xiaoyin Chen, Jiarui Lu, Minsu Kim, Dinghuai Zhang, Jian Tang, Alexandre Piché, Nicolas Gontier, Yoshua Bengio, and Ehsan Kamalloo. Self-evolving curriculum for llm reasoning, 2025. URL https://arxiv.org/abs/2505.14970.

Jorge (Zhoujun) Cheng, Shibo Hao, Tianyang Liu, Fan Zhou, Yutao Xie, Feng Yao, Yuexin Bian, Nilabjo Dey, Yonghao Zhuang, Yuheng Zha, Yi Gu, Kun Zhou, Yuqi Wang, Yuan Li, Richard Fan, Jianshu She, Chengqian Gao, Abulhair Saparov, Taylor W. Killian, Haonan Li, Mikhail Yurochkin, Eric Xing, Zhengzhong Liu, and Zhiting Hu. Revisiting reinforcement learning for llm reasoning from a cross-domain perspective. In *Advances in Neural Information Processing Systems*, volume 38, Main Conference. Curran Associates, Inc., 2025. doi: 10.52202/085713-4479.

Lang Feng, Zhenghai Xue, Tingcong Liu, and Bo An. Group-in-group policy optimization for llm agent training. In *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 46375–46408. Curran Associates, Inc., 2025. doi: 10.52202/085713-1544.

Carlos Florensa, David Held, Markus Wulfmeier, Michael Zhang, and Pieter Abbeel. Reverse curriculum generation for reinforcement learning. In Sergey Levine, Vincent Vanhoucke, and Ken Goldberg (eds.), *Proceedings of the 1st Annual Conference on Robot Learning*, volume 78 of *Proceedings of Machine Learning Research*, pp. 482–495. PMLR, 13–15 Nov 2017. URL https://proceedings.mlr.press/v78/florensa17a.html.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, , et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09422-z. URL http://dx.doi.org/10.1038/s41586-025-09422-z.

Qianyue Hao, Lin Chen, Xiaoqian Qi, Yuan Yuan, Zefang Zong, Hongyi Chen, Keyu Zhao, Shengyuan Wang, Yunke Zhang, Jian Yuan, and Yong Li. Reinforcement learning in the era of large language models: Challenges and opportunities. *ACM Comput. Surv.*, 2026. ISSN 0360-0300.

Zeyu Huang, Tianhao Cheng, Zihan Qiu, Zili Wang, Yinghui Xu, Edoardo M Ponti, and Ivan Titov. Blending supervised and reinforcement fine-tuning with prefix sampling. *arXiv preprint arXiv:2507.01679*, 2025.

Mingjie Liu, Shizhe Diao, Ximing Lu, Jian Hu, Xin Dong, Yejin Choi, Jan Kautz, and Yi Dong. Prorl: Prolonged reinforcement learning expands reasoning boundaries in large language models. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 17998–18031. Curran Associates, Inc., 2025. doi: 10.52202/085713-0608. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/1a22b912945fb7c0bdd079e792b31b6f-Paper-Conference.pdf.

Mingyang Liu, Gabriele Farina, and Asuman Ozdaglar. Uft: Unifying supervised and reinforcement fine-tuning. *Advances in Neural Information Processing Systems*, 38:101347–101383, 2026.

Yi Lu, Jianing Wang, Linsen Guo, Wei He, Hongyin Tang, Tao Gui, Xuanjing Huang, Xuezhi Cao, Wei Wang, and Xunliang Cai. R-horizon: How far can your large reasoning model really go in breadth and depth? In *International Conference on Learning Representations*, volume 2026, pp. 102286–102310, 2026.

Junhyuk Oh, Yijie Guo, Satinder Singh, and Honglak Lee. Self-imitation learning. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 3878–3887. PMLR, 2018.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744, 2022.

QwenTeam. Qwen3.5: Towards native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5.

Cinjon Resnick, Roberta Raileanu, Sanyam Kapoor, Alexander Peysakhovich, Kyunghyun Cho, and Joan Bruna. Backplay: "man muss immer umkehren", 2022. URL https://arxiv.org/abs/1807.06919.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Huaijie Wang, Shibo Hao, Hanze Dong, Shenao Zhang, Yilin Bao, Ziran Yang, and Yi Wu. Offline reinforcement learning for LLM multi-step reasoning. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 8881–8893. Association for Computational Linguistics, 2025. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.464.

Zhiheng Xi, Wenxiang Chen, Boyang Hong, Senjie Jin, Rui Zheng, Wei He, Yiwen Ding, Shichun Liu, Xin Guo, Junzhe Wang, et al. Training large language models for reasoning through reverse curriculum reinforcement learning. *arXiv preprint arXiv:2402.05808*, 2024.

Jianhao Yan, Yafu Li, Zican Hu, Zhi Wang, Ganqu Cui, Xiaoye Qu, Yu Cheng, and Yue Zhang. Learning to reason under off-policy guidance. In *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 117157–117186. Curran Associates, Inc., 2025. doi: 10.52202/085713-3904.

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Yang Yue, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in llms beyond the base model? In *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 57654–57689, 2025. doi: 10.52202/085713-1933.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah Goodman. Star: Bootstrapping reasoning with reasoning. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 15476–15488. Curran Associates, Inc., 2022. doi: 10.52202/068431-1126.

Weihao Zeng, Yuzhen Huang, Qian Liu, Wei Liu, Keqing He, Zejun MA, and Junxian He. SimpleRL-zoo: Investigating and taming zero reinforcement learning for open base models in the wild. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=vSMCBUgrQj.

Haizhong Zheng, Yang Zhou, Brian Bartoldson, Bhavya Kailkhura, Fan Lai, Jiawei Zhao, and Beidi Chen. Act only when it pays: Efficient reinforcement learning for llm reasoning via selective rollouts. In *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 124321–124346. Curran Associates, Inc., 2025. doi: 10.52202/085713-4143.

A Experimental setting

A.1 Datasets

We train our models on a mixture of Countdown and MATH. These datasets provide complementary forms of explicit reasoning supervision. Each training example contains a single ground-truth solution path, which is used to construct supervised guidance and identify the transition from guided to autonomous reasoning; no additional teacher-generated trajectories are introduced. At evaluation time, the model receives only the problem statement, without access to solution paths or hints. We evaluate in-domain generalization on held-out Countdown and MATH problems, and assess cross-benchmark transfer on GSM8K, AMC, AIME, Minerva Math, OlympiadBench, and the decontaminated MATH-500 benchmark. This evaluation design examines whether learning from explicit reasoning paths improves autonomous reasoning on unseen problems, and whether the acquired reasoning ability transfers beyond the training distributions. Table 6 shows the details.

A.2 Baseline method

We compare our method with a set of representative baselines covering different combinations of supervised and reinforcement fine-tuning. **SFT** fine-tunes the base model on the ground-truth reasoning paths, whereas **RFT** directly optimizes the base model with GRPO using verifiable outcome rewards. We additionally include **SimpleRL-Zero** as a representative zero-RL recipe that develops reasoning behavior directly from a base model without supervised fine-tuning. **SFT-RFT** follows the standard sequential pipeline by applying GRPO to the SFT checkpoint. Among unified training approaches, **UFT** uses annealed expert hints together with a hint-likelihood objective, while **Prefix-RFT** combines a ground-truth prefix with an on-policy continuation and optimizes the resulting hybrid trajectory according to its outcome. $\mathbf{R^3}$ uses reference prefixes to initialize generation from intermediate reasoning states and mixes different prefix depths during training. For methods requiring demonstrations, we use only the same ground-truth solution path provided by the training data and introduce no additional teacher-generated trajectories.

A.3 Implementation details

All experiments are conducted using eight NVIDIA A100 GPUs. For controlled comparison, all methods use the same training data, evaluation benchmarks, sequence-length limits, and evaluation protocol. The RL-based methods share the same optimization configuration unless a hyperparameter is an essential part of the original method. In particular, we train for 500 optimization steps with a global prompt batch size of 64 and sample four responses per prompt. All the hyperparameters for training can be found in Table 7.

B Algorithm

To provide a quick overview of our method, we present the pseudocode for CRL (Algorithm 1).

Table 6: Statistics of the test datasets.

Dataset	Test samples	Min tokens	Max tokens
AIME	30	13	13
AMC	83	13	16
MATH500	500	11	41
Minerva	272	10	58
OlympiadBench	673	11	121
GSM8K	1,319	38	427
MATH	1,024	38	2,018
Countdown	1,024	52	90

Table 7: Training configuration and reward settings.

Parameter	**Value**
Training	
Training batch size	64
Validation batch size	256
Mini-batch size	16
Prefix length schedule	$95\% \rightarrow 5\%$; removed after the curriculum
Infix length	10%
Optimizer	AdamW
Learning rate	1×10^{-6}
Training steps	500
Curriculum Steps	300
Policy-update epochs	1
Rollouts per prompt	4
ω	0.001
λ_{KL}	0.001
λ_{ent}	0.001
Maximum prompt length	1,024 tokens
Maximum response length	1,024 tokens
Reward	
Accuracy reward	1.0
Format correctness reward	0.1
Incorrect reward	0.0

C CASE STUDIES

We have provided representative samples as case studies. These four cases illustrate distinct errors in constraint satisfaction, quantitative reasoning, algebraic manipulation, and the application of geometric properties. All the cases are generated using fine-tuning Qwen3.5-0.8B.

In Case 1, GRPO repeatedly tries different arithmetic combinations but ultimately fails to construct a valid reasoning path: its final expression reaches 67 by using 68 twice and omitting 24. This repeated but unsuccessful search provides qualitative support for our motivation to introduce effective supervisory guidance during RL, encouraging coherent reasoning that satisfies task constraints. **In Case 2**, GRPO fails to account correctly for the combined capacity of two trays and makes a subtraction error. **In Case 3**, both the coefficient obtained by combining the fractions and the subsequent multiplication are incorrect. **In Case 4**, although GRPO correctly identifies the hypotenuse as the diameter of the circumcircle, it incorrectly assigns a radius of 2 to the first circle, leading to an incorrect area ratio and final answer. By comparison, CRL transforms problem conditions into explicit and verifiable intermediate steps, thereby correctly constructing a reasoning path that meets the requirements. The specific details of these cases are as follows:

System Prompt. A conversation between User and Assistant. The user asks a question, and the Assistant solves it. The assistant first thinks about the reasoning process in the mind and then provides the user with the answer. User: $\{question\}$ Show your work in $< think >< /think >$

Algorithm 1 Curriculum Reinforcement Learning (CRL)

Input: Reference dataset $\mathcal{D}_{\mathrm{SL}}$, initial policy $\pi_{\boldsymbol{\theta}}$, fixed reference policy π_{ref}, and task verifier.
Parameters: Training steps K, curriculum steps K_{curr}, group size G, annealed prefix schedule $\{\mathcal{P}_k^{\mathrm{hint}}\}$, and base weights ω.
Output: Trained policy $\pi_{\boldsymbol{\theta}}$.

for $k = 1, \ldots, K$ **do**
  $\pi_{\mathrm{old}} \leftarrow \pi_{\boldsymbol{\theta}}$; sample a problem batch $\mathcal{B} \subset \mathcal{D}_{\mathrm{SL}}$; $\mathcal{Z} \leftarrow \varnothing$
  for each $(\mathbf{X}, \mathbf{Y}^\star) \in \mathcal{B}$ **do**
    if $k \le K_{\mathrm{curr}}$ **then**
      Sample $\ell \sim \mathcal{P}_k^{\mathrm{hint}}$ and extract reference prefix $\mathbf{H}_\ell$
    else
      $\mathbf{H}_\ell \leftarrow \varnothing$
    end if
    $\mathbf{S}_\ell \leftarrow \mathbf{X} \oplus \mathbf{H}_\ell$; $f_\ell \leftarrow |\mathbf{H}_\ell|/|\mathbf{Y}^\star|$
    Sample $\mathbf{Y}^i \sim \pi_{\mathrm{old}}(\cdot \mid \mathbf{S}_\ell)$, $i = 1, \ldots, G$
    Obtain verifier rewards $\{r_i\}_{i=1}^G$ and rollout-success indicators $\{c_i\}_{i=1}^G$
    $\hat{p}_\ell \leftarrow G^{-1} \sum_{i=1}^G c_i$; compute GRPO advantages $\{A_i\}_{i=1}^G$ from $\{r_i\}_{i=1}^G$
    if $k \le K_{\mathrm{curr}}$ **then**
      $\omega_\ell^{\mathrm{hint}} \leftarrow \mathrm{sg}[\omega\,(1 - (1 - f_\ell)\hat{p}_\ell)]$
      $\omega_i^{\mathrm{infix}} \leftarrow \mathrm{sg}[\omega\, c_i(1 - \hat{p}_\ell)]$, $i = 1, \ldots, G$
      Select infix tokens in each $\mathbf{Y}^i$ to construct $m_{i,t}^{\mathrm{infix}}$
    else
      $\omega_\ell^{\mathrm{hint}} \leftarrow 0$; $\omega_i^{\mathrm{infix}} \leftarrow 0$; $m_{i,t}^{\mathrm{infix}} \leftarrow 0$ for all i, t
    end if
    Store the problem, prefix, continuations, advantages, weights, and masks in $\mathcal{Z}$
  end for
  Optimize $\boldsymbol{\theta}$ on $\mathcal{Z}$ using $\mathcal{L}_{\mathrm{RL}} + \mathcal{L}_{\mathrm{hint}} + \mathcal{L}_{\mathrm{infix}}$, where
  $\mathcal{L}_{\mathrm{hint}} = -\mathbb{E}_{\mathrm{hint}}\big[\omega_\ell^{\mathrm{hint}} \log \pi_{\boldsymbol{\theta}}(h_{\ell,t} \mid \mathbf{X}, \mathbf{H}_{\ell,<t})\big]$
  $\mathcal{L}_{\mathrm{infix}} = -\mathbb{E}_{\mathrm{resp}}\big[\omega_i^{\mathrm{infix}} m_{i,t}^{\mathrm{infix}} \log \pi_{\boldsymbol{\theta}}(y_t^i \mid \mathbf{S}_\ell, \mathbf{Y}_{<t}^i)\big]$
end for
return $\pi_{\boldsymbol{\theta}}$

tags. And return the final answer in $< answer >< /answer >$ tags, for example $< answer >$ 12 $< /answer >$. Assistant: Let me solve this step by step.

- **Case 1**

  Question: Using the numbers $[68, 46, 24, 23]$, create an equation that equals 67. You can use basic arithmetic operations $(+, -, *, /)$ and each number can only be used once.

  GRPO Response

  $< think >$ 68 - 24 = 44 46 - 44 = 2 2 + 23 = 25 Not yet 67. Let's try another combination.
  68 - 23 = 45 45 + 24 = 69 69 - 46 = 23 Still too small. Let's look closer again.
  68 - 46 = 22 22 * 24 = 528 528 / 23 = 23 Still too small. Let's try another combination.
  68 + 24 = 92 92 - 23 = 69 69 - 46 = 23 Let's look closer carefully again.
  68 - 22 = 46 46 + 23 = 69 69 - 68 = -1 Let's try another.
  68 / 24 = 2.875 2.875 * 23 = 66.125 Not an integer. Let's look closer again.
  68 - 23 = 45 45 + (68 - 46) = 45 + (22) = 67
  So the final equation is ((68 - 23) + (68 - 46)) $< /think >$
  $< answer >$((68 - 23) + (68 - 46)) $< /answer >$

  CRL(Ours) Response

  $< think >$ We have the numbers [68, 46, 24, 23]. We need to get 67.
  Try: 68 + 46 = 114. 114 - 24 = 90. 90 - 23 = 67.
  Yes, that works: ((68 + 46) - 24) - 23.
  Check: (68 + 46) = 114 114 - 24 = 90 90 - 23 = 67 Yes, it's correct.$< /think >$
  $< answer >$((68 + 46) - 24) - 23 $< /answer >$

- **Case 2**

Question: Jaime places eggs on the tray. Each tray can hold 24 eggs. If he has 64 eggs and 2 trays, how many eggs won't he be able to place on the tray?

GRPO Response

$< think >$ 64 eggs - 24 eggs/tray = 28 eggs left over if we use 2 trays.

Therefore, he won't be able to place 28 eggs on the trays. $< /think >$

$< answer > 28 < /answer >$

CRL(Ours) Response: $< think >$ We have 64 eggs and 2 trays, each holding 24 eggs. Total capacity = 2 trays * 24 eggs/tray = 48 eggs.

We have 64 eggs, so the number of eggs that can be placed = 48.

The number of eggs that won't be placed = 64 - 48 = 16. Therefore, the answer is 16.$< /think >$

$< answer > 16 < /answer >$

- **Case 3**

Question: Solve for x: $\frac{x}{3} + \frac{x}{4} = 14$.

GRPO Response

$< think >$ We have $\frac{5x}{12} = 14$

$x = 14 * \frac{12}{5} = 8.4 < /think >$

$< answer > 8.4 < /answer >$

CRL(Ours) Response

$< think >$ We have the equation: $\frac{x}{3} + \frac{x}{4} = 14$.

Combine the left side: $\frac{x}{3} + \frac{x}{4} = x\left(\frac{1}{3} + \frac{1}{4}\right)$.

Find the common denominator for the fractions: $\frac{1}{3} + \frac{1}{4} = \frac{4}{12} + \frac{3}{12} = \frac{7}{12}$.

So, $x \cdot \frac{7}{12} = 14$. $x = 14 \cdot \frac{12}{7} = 2 \cdot 12 = 24$.

Therefore, $x = 24$. $< /think >$

$< answer > 24 < /answer >$

- **Case 4**

Question: A $3-4-5$ right triangle is inscribed in circle A, and a $5-12-13$ right triangle is inscribed in circle B. Find the ratio of the area of circle A to the area of circle B. The final answer can be written in the form $\frac{m}{n}$, where m and n are relatively prime positive integers. What is $m+n$?

GRPO Response

$< think >$ The $3-4-5$ right triangle is a right triangle whose hypotenuse is a diameter of circle A, so circle A has radius 2 and area $\pi(2^2) = 4\pi$.

The $5-12-13$ right triangle is a right triangle whose hypotenuse is a diameter of circle B, so circle B has radius 6.5 and area $\pi(6.5^2) = 42.25\pi$.

Thus the ratio of the area of circle A to the area of circle B is $\frac{4\pi}{42.25\pi} = \frac{4}{42.25} = \frac{16}{169}$, and $m+n = 16+169 = 185$. $< /think >$

$< answer > 185 < /answer >$

CRL(Ours) Response

$< think >$ We have two right triangles inscribed in circles.

1. A $3-4-5$ right triangle is inscribed in circle A. For a right triangle with sides a, b, c (with c being the hypotenuse), the circumcenter is the midpoint of the hypotenuse. So, for $3-4-5$, the hypotenuse is 5. The circumradius $R_A = \frac{5}{2}$. The area of circle A is $S_A = \pi R_A^2 = \pi\left(\frac{5}{2}\right)^2 = \pi\frac{25}{4} = \frac{25\pi}{4}$.

2. A $5-12-13$ right triangle is inscribed in circle B. Similarly, the hypotenuse is 13. The circumradius $R_B = \frac{13}{2}$. The area of circle B is $S_B = \pi R_B^2 = \pi\left(\frac{13}{2}\right)^2 = \pi\frac{169}{4} = \frac{169\pi}{4}$.

The ratio of the areas $S_A/S_B = \frac{25\pi/4}{169\pi/4} = \frac{25}{169}$. So, $m = 25, n = 169$. $m+n = 25+169 = 194$. The answer is 194. $< /think >$

$< answer > 194 < /answer >$

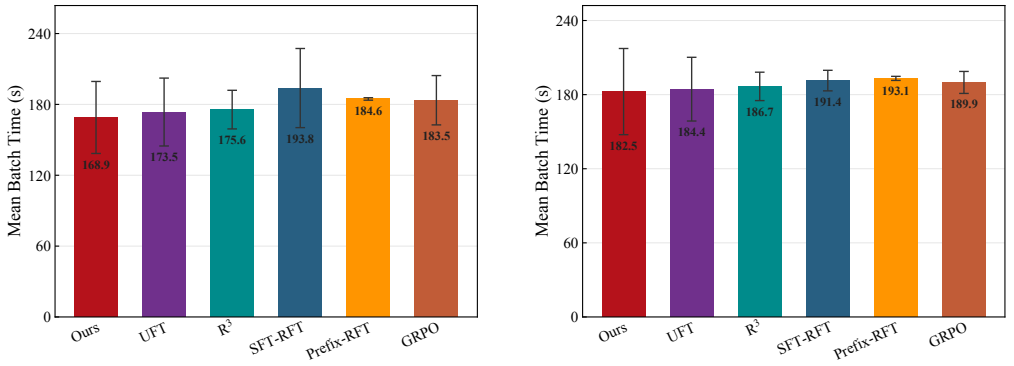

Figure 8: Mean Training Batch Time including training updates: Qwen3.5-0.8B(left) and Qwen3.5-2B(right). The time spent on SFT within the SFT-RFT is not included in the statistics.