CAN LLMS LEARN TO REASON ROBUSTLY UNDER NOISY LABELS?

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement Learning with Verifiable Rewards (RLVR) effectively trains reasoning models that rely on abundant perfect supervision, but its vulnerability to unavoidable noisy labels remains underexplored. Through empirical validation, we find that the vanilla RLVR method demonstrates degraded performance for two reasons: (i) **data waste**: when the LLM itself cannot roll out the noisy label, many data points are overlooked in training; (ii) **policy misguidance**: if the wrong label is rolled out, the policy will be guided toward the wrong direction, hurting model training. Fortunately, by inspecting the training dynamics, we identify a **Early Correctness Coherence** phenomenon: *in the early stage of training, the probability of correct answers rises similarly for clean and noisy samples*. Inspired by this, we develop a novel **O**nline **L**abel **R**efinement (**OLR**) algorithm that performs self-consistent label correction when a sample's predicted labels are historically stable and increasingly confident. Evaluated on six in-distribution (ID) math benchmarks and three out-of-distribution (OOD) tasks under 0.1–0.9 noise ratios, OLR achieves average gains of 3.6% (ID) / 3.3% (OOD) under inactive noise and 3.9% (ID) / 4.6% (OOD) under active noise.

1 INTRODUCTION

Following the success of DeepSeek-R1 (Guo et al., 2025), reinforcement learning with verifiable rewards (RLVR) has rapidly become a powerful paradigm for training reasoning-oriented language models. Algorithms like GRPO (Shao et al., 2024) and its variants (Liu et al., 2025; Zheng et al., 2025; Yu et al., 2025) remove the need for critic models via intra-group advantage estimation, but still depend on high-quality reference labels. In practice, expert scarcity and task complexity make perfect annotation infeasible, introducing inevitable label noise. While some studies attribute RLVR's robustness to pre-trained priors (Shao et al., 2025; Lv et al., 2025), others reveal its limitations and show that label noise can distort optimization, causing policy collapse and performance degradation beyond the reach of current methods (Rad et al., 2026; Egashira et al., 2026; Zhu & Kang, 2026). Since reference labels anchor policy updates, understanding how noise affects learning and developing robust mitigation strategies are critical for RLVR's reliability and scalability, yet substantial room remains for more resilient approaches.

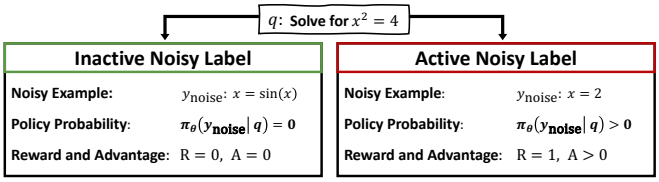


Figure 1: **Two types of noisy labels in RLVR.** *Inactive* (left): incorrect labels that the model cannot realize or reinforce, thus remaining inactive. *Active* (right): incorrect labels that the model can realize and reinforce, thus remaining active.

To understand the impact of noisy labels in RLVR, we conduct an empirical study and find that the vanilla RLVR method demonstrates degraded performance for two reasons: (i) **data waste**: when the LLM itself cannot roll out any answer that matches the noisy label, many data points are overlooked in training. We refer to this as an *inactive noisy label*; (ii) **policy misguidance**: if the wrong label is matched, the policy will be guided toward the wrong direction, severely hurting model training. We refer to this as an *active noisy label*. As shown in Fig.2, both types of noisy samples can result in wasted training cycles and reduced performance; more severely, only **10%** of active noisy labels can reduce the model's average performance by **6.3%**.

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.

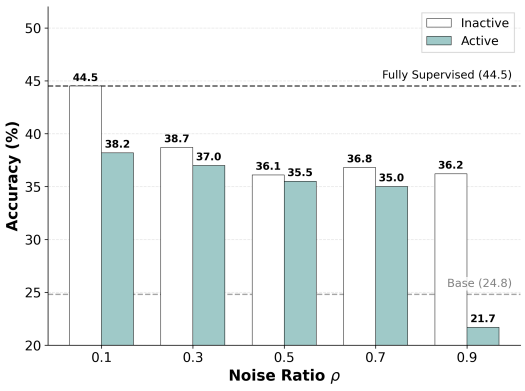


Figure 2: Performance degradation under noisy labels: data waste (Inactive) and policy misguidance (Active).

To investigate how noisy labels affect RLVR training and the opportunity to improve robustness, we examine the learning dynamics and uncover a distinctive phenomenon. Interestingly, we observe a key dynamic during training with noisy labels, which we term ***Early Correctness Coherence***: although noisy samples lag in later stages, the probability of correct answers rises similarly for clean and noisy samples in early training (Fig. 3). This indicates that training already yields correct answers for noisy samples, and this phenomenon can be used to detect noisy labels: *a sample whose predicted probability for the initial label fails to rise, or declines, is likely noisy and should be corrected.* How, then, can noisy labels be corrected? Recent work (Yue et al., 2025) suggests that RLVR may not actually exceed the capability upper bound of the base pretrained model, but rather increases the probability that the model outputs correct answers. In other words, the correct answers are already encoded in the base model's outputs, so we can extract potentially correct labels from these outputs. A naive solution is to adopt majority answers (Zuo et al., 2025) by voting among rollouts as a replacement. Yet this approach often fails early in training as the majority answer tends to be wrong, which can easily lead to model collapse (He et al., 2026). However, we argue that a truly reliable majority answer should also satisfy the Early Correctness Coherence phenomenon. That is, as training progresses and the probability of outputting the correct answer increases, majority answers with steadily rising probabilities are much more likely to be genuinely correct. Thus, we can use such majority answers to correct potential initial noisy labels.

Motivated by the observation above, we propose **O**nline **L**abel **R**efinement (**OLR**) for denoising. Instead of relying solely on initial labels, OLR selectively corrects potentially noisy labels using policy-generated majority answers, which increasingly reflect correctness. Intuitively, this resembles human learning: early coherence fosters reasoning ability, and answers that repeatedly emerge with growing probabilities are more likely correct. This intuition is formalized by monitoring two signals: ① `Positive Convergence Slope`. A positive convergence slope indicates that as the model repeatedly attempts the problem, more rollouts converge to the same answer, suggesting it provides a positive expected advantage and meaningful learning signal as the policy improves. ② `Historical Consistency`. If the same majority answer remains dominant across updates, transient or accidental answers are filtered out, avoiding misleading corrections.

In summary, as reasoning improves, noisy labels are gradually overwritten, allowing the policy to refine its own targets. We evaluate OLR on six in-distribution (ID) tasks (AIME24/25, AMC, MATH-500, Minerva, Olympiad) and three out-of-distribution (OOD) tasks (ARC-c, GPQA-diamond, MMLU-Pro). Across noise ratios 0.1–0.9, OLR yields consistent gains: under inactive noise, average improvements of **3.6%** (ID) and **3.3%** (OOD); under active noise, **3.9%** and **4.6%**, respectively. These results show OLR robustly handles both noise types across diverse tasks.

2 METHOD

Section 2.1 formulates the standard RLVR problem and provides the definition of noisy labels in RLVR. Section 2.2 presents a theoretical analysis of the Early Correctness Coherence phenomenon observed in practice. Building on this analysis, Section 2.3 introduces our proposed method, **O**nline **L**abel **R**efinement (**OLR**), a noisy label mitigation approach for RLVR.

2.1 PROBLEM FORMULATION

We study reinforcement learning with verifiable rewards (RLVR), where a policy $\pi_\theta(y \mid x)$ generates solutions y for prompts x. For each x, the model samples K rollouts $\mathcal{Y}(x) \sim \pi_\theta(\cdot \mid x)$, and a verifier provides scalar rewards $r(x, y)$, typically binary. Group Relative Policy Optimization (GRPO) (Shao et al., 2024) normalizes rewards within rollout groups to compute advantages: $A(x, y^{(k)}) =$

$\frac{r(x,y^{(k)})-\mu(x)}{\sigma(x)+\epsilon}$, with $\mu(x)$ and $\sigma(x)$ denoting group mean and standard deviation. The policy is updated by maximizing the clipped advantage-weighted likelihood with KL regularization:

$$\mathcal{L}_{\text{GRPO}}(\theta) = \mathbb{E}_{x,y}\Big[\text{CLIP}\Big(A(x,y)\frac{\pi_\theta(y \mid x)}{\pi_{\theta_{\text{old}}}(y \mid x)}, \epsilon\Big) - \beta \cdot \mathbb{D}_{\text{KL}}[\pi_\theta \| \pi_{\text{ref}}]\Big]. \tag{1}$$

While RLVR methods are effective, they assume clean labels, yet noisy labels are common. We denote initial annotations as $\tilde{y}(x)$ and ground truth as $y^\star(x)$. Whether noisy labels are reinforced depends on the policy's ability to roll them out, a property termed Rollout Feasibility.

Definition 2.1 (Rollout Feasibility). *For K rollouts $\mathcal{Y}_t^{(K)}(x)$, define $F_t^{(K)}(x,y) = \mathbb{I}[y \in \mathcal{Y}_t^{(K)}(x)]$. A label is rollout-feasible when $F_t^{(K)}(x,y) = 1$; this status varies with K and training step t.*

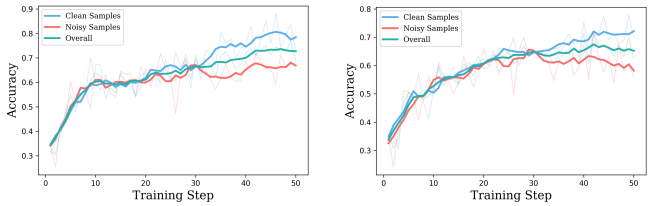
Definition 2.2 (Inactive and Active Noisy Labels). *At step t, a wrong reference label $\tilde{y}(x) \neq y^\star(x)$ is inactive if $F_t^{(K)}(x,\tilde{y}) = 0$, and active if $F_t^{(K)}(x,\tilde{y}) = 1$.*

This classification is step-wise: if the correct answer is absent, the current group cannot reinforce it, though later groups may.

This formalization raises a key question: *can models learn robustly reasoning under noisy labels?* We address this by analyzing training dynamics and finding the Early Correctness Coherence.

2.2 Early Correctness Coherence Phenomenon in Noisy RLVR

We analyze the training dynamics under both inactive and active noisy labels and find that, despite their different effects on training, they exhibit a remarkably similar trajectory. As shown in Figure 3, the dynamics unfold in two stages: *(i) in the early stage, the probability of correct answers rises similarly for clean and noisy samples; (ii) in the later stage, noisy samples begin to lag behind.* We term this phenomenon ***Early Correctness Coherence***. This interesting pattern indicates that, despite noisy supervision, the model has the opportunity to produce partially correct answers for noisy samples at the beginning of training. A natural interpretation is that early training first captures reasoning patterns that are relatively easy to learn. Updates from clean samples strengthen these shared reasoning structures, which in turn benefit noisy samples that rely on the same parameters. In other words, the model's progress on clean data inadvertently transfers to noisy data, allowing both groups to improve together before noise begins to dominate. This positive cross-sample coupling explains why noisy samples initially "ride along" with clean ones, even though their labels are incorrect.



(a) Inactive Noise (b) Active Noise

Figure 3: **Early Correctness Coherence** Phenomenon. Training accuracy of Qwen3-4B-Base (Yang et al., 2025a) under noisy supervision (noise ratio 0.5). For each sample, we take the majority answer as the model's prediction. Clean and noisy samples exhibit similar learning dynamics early in training but gradually diverge as training progresses.

We formalize this mechanism by analyzing the dynamics of the log-ratio between correct and noisy labels. Under mild assumptions, initial correctness bias, positive gradient coupling between clean and noisy samples, and bounded mean drift, we prove the following theorem.

Theorem 2.3 (Early Correctness Coherence in Noisy RLVR). *(Informal) Let $\mathcal{D} = \mathcal{D}_{\text{clean}} \cup \mathcal{D}_{\text{noise}}$ with noise ratio $\rho = |\mathcal{D}_{\text{noise}}|/|\mathcal{D}|$. For each prompt x at epoch t, let $p_t(y|x) = \pi_{\theta_t}(y|x)$ and define log-ratio $L_t(x) = \log \frac{p_t(y^\star(x)|x)}{p_t(\tilde{y}(x)|x)}$. The log-ratio analysis assumes both probabilities are positive; unlike the finite-K categories above, it does not require either answer to have exactly zero policy probability. Assume: (i) $p_0(y^\star) > p_0(\tilde{y})$; (ii) positive cross-sample coupling between the clean sample x_c and the noisy sample x_n:*

$$\mathbb{E}_{x_c,x_n}\Big[\nabla_\theta \log \pi(y^\star(x_c)|x_c) \cdot \nabla_\theta \log \pi(y^\star(x_n)|x_n)\Big] \geq \gamma > 0\,; \tag{2}$$

(iii) mean drift $\Delta_s = \gamma(1-\rho)G_c - \rho G_n$*, where* G_c *and* G_n *are the mean clean and noisy advantage magnitudes. Let* $\rho_c = \frac{\gamma G_c}{\gamma G_c + G_n}$*. If* $\rho < \rho_c$ *and* $K \gtrsim \log(T/\delta)$*, then with probability at least* $1-\delta$*, for all* $t \le T$*,*

$$L_t(x) \ge L_0(x) + \eta t \left(\Delta_s - O\left(\sqrt{\frac{\log(T/\delta)}{K}} \right) \right), \tag{3}$$

implying that $p_t(y^\star(x)|x) \gg p_t(\tilde{y}(x)|x)$.

See Appendix B for the formal Theorem B.1 and the full proof, and Appendix Figure 8 for experimental verification of Theorem 2.3's assumptions.

Implication. Theorem 2.3 shows that in the early phase, correct answers emerge on noisy samples with increasing probability due to cross-sample coupling. This insight motivates our method: by recovering these emergent correct answers from model rollouts, we can detect and refine noisy labels.

2.3 Online Label Refinement (OLR)

Motivated by **Early Correctness Coherence**, OLR proposes candidate labels from on-policy rollout majorities. A rising majority pass rate alone does not establish that a particular answer is correct, because the majority identity can change between visits. OLR therefore checks both increasing rollout concentration and historical majority agreement before replacing a reference label.

Formally, at epoch t, the policy generates K rollouts $\mathcal{Y}_t(x) = \{y_t^{(1)}, \dots, y_t^{(K)}\} \sim \pi_\theta(\cdot \mid x)$. To estimate solution probabilities, we use the empirical pass rate from these rollouts. Since enumerating all possible answers is infeasible, we track only the *majority answer* $y_t^{\text{maj}}(x) = \arg\max_c |\{y \in \mathcal{Y}_t(x) : y = c\}|$, with pass rate $p_t^{\text{maj}}(x) = \frac{1}{K}|\{y \in \mathcal{Y}_t(x) : y = y_t^{\text{maj}}(x)\}|$. We maintain a trajectory $\mathcal{H}_t(x) = \{(t', y_{t'}^{\text{maj}}(x), p_{t'}^{\text{maj}}(x)) : t' \in [1, t]\}$ to evaluate the reliability of the candidate's answer.

Algorithm 1 Online Label Refinement

Require: Dataset $D = \{(x, \tilde{y}(x))\}$, policy π_θ, rollouts K, early epoch T, slope threshold δ_{slope}

Set initial training label $\hat{y}_t(x) \leftarrow \tilde{y}(x)$
for epoch $t = 0, 1, 2\dots$ **do**
  for prompt $x \in D$ **do**
    Generate K rollouts and majority answer $y_t^{\text{maj}}(x)$ with pass rate $p_t^{\text{maj}}(x)$
    Compute $S_t(x)$ and $y_t^{\text{hist}}(x)$
    if $t > T$ **then**
      if $S_t(x) > \delta_{\text{slope}}$ and $y_t^{\text{maj}}(x) = y_t^{\text{hist}}(x)$ **then**
        $\hat{y}_t(x) \leftarrow y_t^{\text{maj}}(x)$
      else
        $\hat{y}_t(x) \leftarrow \tilde{y}(x)$
      end if
    end if
  end for
  Update policy π_θ with GRPO using $\{(x, \hat{y}_t(x))\}$
end for

Criterion ①: Positive Convergence Slope. To summarize how concentrated the rollout groups become over visits, we compute the linear regression slope of their majority pass-rate trajectory. Let $\mathbf{p}_t(x) = [p_1^{\text{maj}}(x), \dots, p_t^{\text{maj}}(x)]^\top$ and $\mathbf{t} = [1, \dots, t]^\top$. The slope is

$$S_t(x) = \frac{(\mathbf{t} - \bar{t}\mathbf{1})^\top (\mathbf{p}_t(x) - \bar{p}\mathbf{1})}{(\mathbf{t} - \bar{t}\mathbf{1})^\top (\mathbf{t} - \bar{t}\mathbf{1})}, \tag{4}$$

where $\bar{t} = \frac{1}{t}\sum_{i=1}^t i$ and $\bar{p} = \frac{1}{t}\sum_{i=1}^t p_i^{\text{maj}}(x)$. A positive slope $S_t(x) > \delta_{\text{slope}}$ indicates that the pass rate of each step's winning answer tends to increase. It does not track the probability of one fixed answer or, by itself, establish correctness. The historical-consistency criterion below separately checks the candidate's identity.

Criterion ②: Historical Consistency. However, the majority answer may be volatile. To filter stochastic fluctuations, we check whether the current majority answer matches the historical majority $y_t^{\text{hist}}(x) = \arg\max_y |\{(t', y_{t'}^{\text{maj}}(x)) \in \mathcal{H}_t(x) : y_{t'}^{\text{maj}}(x) = y\}|$. The consistency indicator is

$$C_t(x) = \mathbb{I}\left(y_t^{\text{maj}}(x) = y_t^{\text{hist}}(x) \right). \tag{5}$$

After an initial early learning phase of T epochs to accumulate rollout statistics, the effective label $\hat{y}_t(x)$ used for reward computation is

$$\hat{y}_t(x) = \begin{cases} y_t^{\text{maj}}(x) & \text{if } S_t(x) > \delta_{\text{slope}} \text{ and } C_t(x), \\ \tilde{y}(x) & \text{otherwise.} \end{cases} \tag{6}$$

We adopt GRPO as the base optimizer. The following result gives conditions under which OLR reduces effective label noise; Fig. 5 (f) evaluates the effect during training.

Theorem 2.4 (Conditional Noise-Rate Reduction under OLR)**.** *At a fixed update, let ρ be the original fraction of incorrect reference labels. For originally noisy and clean prompts, let q_n, q_c be their replacement probabilities and e_n, e_c the error probabilities of the replacement answer conditional on selection, respectively. Then the expected effective noise rate is*

$$\rho_{\text{eff}} = \rho(1 - q_n + q_n e_n) + (1 - \rho) q_c e_c. \tag{7}$$

Thus OLR reduces label noise precisely when $\rho q_n(1 - e_n) > (1 - \rho) q_c e_c$. A larger tolerable original noise rate follows only if the relevant learning criterion depends on ρ_{eff} and this reduction holds at its baseline threshold; neither follows from an unconditioned majority-vote error bound alone.

For a detailed proof, please refer to Appendix B.8. Overall, OLR dynamically selects reliable majority answers to rewrite noisy labels by analyzing the slope of the pass rate trend for majority answers and their historical consistency, thereby achieving effective denoising. Compared to model inference and update, the additional time overhead introduced by OLR is almost negligible (see Appendix Table 4). Below, we demonstrate OLR's effectiveness through experimental results.

3 Experiment

Section 3.1 introduces the design of the experimental datasets, the evaluation benchmarks, and the baseline methods. In Section 3.2, we analyze the performance gains of OLR under various noise ratios, compare it with baselines, and evaluate it in more realistic scenarios, while further examining the effects of problem difficulty and training dynamics, as well as conducting time complexity analysis, parameter sensitivity analysis, and ablation studies. Appendix D presents additional experiments: model scales and types (D.1), theoretical validation (D.4), and training dynamics (D.5).

3.1 Setup

Dataset and Splits Following recent studies (Qu et al., 2026; Zhan et al., 2025) showing that moderately difficult questions are most effective for RLVR training, we adopt this guideline to control confounding factors and accurately assess the impact of noisy labels. Specifically, we use Qwen3-4B-Base to perform inference on DAPO-Math, sampling each example 8 times and computing the pass rate to quantify problem difficulty, with lower pass rates indicating higher difficulty. We then investigate noisy-label effects under two scenarios. Specifically,

- For `High-Utilization Scenario`, we construct a dataset consisting of samples with a pass rate of 0.5. This setting is designed to eliminate the impact of problem difficulty and isolate the most informative samples for RLVR training, where every sample plays a pivotal role in training dynamics. By doing so, we aim to investigate the effect of noisy labels on RLVR training in a controlled environment. In this scenario, we conduct a comprehensive analysis under different noise ratios (0.1, 0.3, 0.5, 0.7, 0.9) and compare the results with various baseline methods.

- For `Realistic Mixed-Difficulty Scenario`, we construct a dataset by collecting samples with pass rates between 0.25 and 0.75. This setup is intended to better reflect the broad range of training sample difficulties found in real-world scenarios, allowing us to investigate the impact of noisy labels under realistic conditions. In practice, since samples are often curated, the proportion of noisy labels is typically not very high, so we analyze the effects at noise ratios of 0.2 and 0.5.

For inactive noise injection, we replace correct labels with strings that the model is unlikely to generate. For active noise, fixed wrong labels are possible, but their effective activity depends on the current policy and K. We use on-policy wrong answers to control the active rate in the main experiments and report fixed-label results in Appendix D.2. For detailed dataset construction and noise injection methods, please refer to Appendix C.1 and C.2 for implementation details.

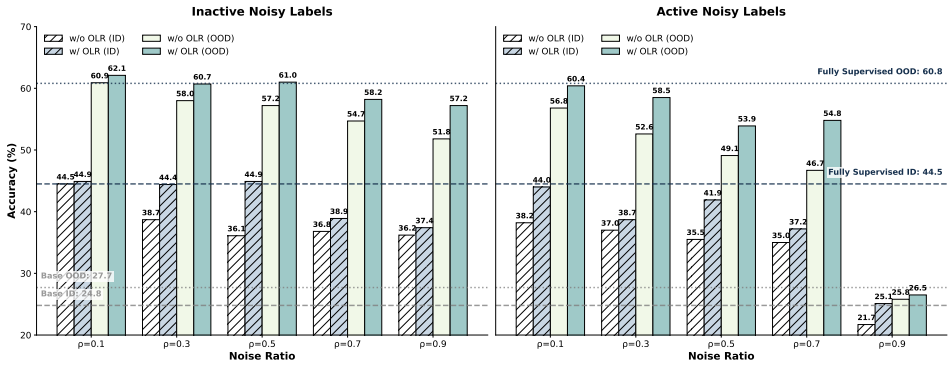


Figure 4: Results on Qwen3-4B-Base under the High-Utilization Scenario with different noise ratios.

Table 1: Results under the High-Utilization Scenario across different baseline methods. **Bold**: inactive-noise best; underline: active-noise best.

Method	In-Distribution							Out-of-Distribution			
	AIME24	AIME25	AMC	MATH-500	Minerva	Olympiad	**Avg.**	ARC-c	GPQA*	MMLU-Pro	**Avg.**
Qwen3-4B-Base	9.6	4.2	34.2	52.6	19.5	28.4	24.8	35.8	14.1	33.3	27.7
Fully Supervised	20.4	17.9	52.9	85.0	42.6	48.1	44.5	86.9	36.4	59.1	60.8
Unsupervised Methods											
TTRL (Zuo et al., 2025)	12.1	8.3	48.2	76.4	36.0	40.3	36.9	**86.9**	30.3	56.0	57.7
Co-reward (Zhang et al., 2025b)	12.1	7.1	44.9	76.0	38.9	40.7	36.6	86.1	29.3	55.8	57.1
Self-certainty (Zhao et al., 2025)	5.8	5.0	28.9	41.8	21.7	20.6	20.6	24.3	12.1	30.6	22.3
Token-entropy (Agarwal et al., 2025)	3.8	2.1	22.3	44.2	15.8	15.7	17.3	17.0	7.6	28.9	17.8
Seq-entropy (Agarwal et al., 2025)	4.6	2.1	25.2	41.0	19.5	19.4	18.6	18.6	6.6	28.4	17.9
Active Noisy Label											
GRPO (Shao et al., 2024)	10.4	8.8	45.8	74.6	33.1	40.4	35.5	73.9	24.7	48.6	49.1
w/ Confidence Penalty (Pereyra et al., 2017)	12.5	9.6	47.0	76.4	34.2	39.6	36.6	82.9	32.3	53.3	56.2
w/ Label Smoothing (Lukasik et al., 2020)	14.6	7.1	44.1	73.2	33.5	38.1	35.1	55.5	24.2	47.4	42.4
w/ Samll-loss Select (Gui et al., 2021)	6.7	1.7	22.4	43.8	16.5	17.6	18.1	32.3	9.6	26.7	22.9
w/ Random Select	9.2	8.8	43.7	74.6	33.1	39.6	34.8	45.6	14.6	46.3	35.5
w/ OLR (ours)	20.4	15.4	49.7	81.4	36.4	48.1	41.9	78.5	28.3	54.8	53.9
Inactive Noisy Label											
GRPO (Shao et al., 2024)	12.5	5.0	45.3	76.0	34.9	42.7	36.1	86.7	28.8	56.0	57.2
w/ Confidence Penalty (Pereyra et al., 2017)	13.3	7.9	48.6	76.6	39.0	43.1	38.1	87.4	31.3	56.6	58.4
w/ Label Smoothing (Lukasik et al., 2020)	11.7	7.9	47.3	78.8	39.0	42.8	37.9	86.5	29.8	57.4	57.9
w/ Samll-loss Select (Gui et al., 2021)	12.1	9.6	46.4	76.8	36.4	38.5	36.6	73.9	30.3	50.7	51.6
w/ Random Select	11.3	12.1	47.4	77.8	38.2	41.3	38.0	78.1	31.8	55.0	55.0
w/ OLR (ours)	**21.7**	**18.3**	**53.9**	**84.2**	**42.3**	**48.9**	**44.9**	86.8	**38.4**	**57.8**	**61.0**

Evaluation We use six in-distribution math reasoning benchmarks: AIME 2024, AIME 2025, AMC (Li et al., 2024), Minerva (Lewkowycz et al., 2022), OlympiadBench (He et al., 2024), and MATH-500 (Hendrycks et al., 2021). We report `avg@32` on AIME 2024/2025 and AMC (due to small test sets) and `pass@1` on the others. For out-of-distribution generalization, we use ARC-c (Clark et al., 2018), GPQA-diamond (Rein et al., 2024) (denoted GPQA*), and MMLU-Pro (Wang et al., 2024a), covering open-domain reasoning, graduate-level science, and academic reasoning. For BIRD Text2SQL (Zhu & Kang, 2026), we use paired original and corrected SQL labels and report execution accuracy on a disjoint, human-corrected test set; details appear in Appendix D.3.

Competitors We compare two categories of baseline methods. The first category consists of unsupervised approaches that do not rely on ground-truth labels: (1) **TTRL** (Zuo et al., 2025); (2) **Co-Reward** (Zhang et al., 2025b); (3) **Self-Certainty** (Zhao et al., 2025); (4) **Token-Level Entropy** (Agarwal et al., 2025); and (5) **Sentence-Level Entropy** (Agarwal et al., 2025). The second category includes noise-robust learning methods and transfer-friendly regularization techniques: (6) **Confidence Penalty** (Pereyra et al., 2017); (7) **Label Smoothing** (Lukasik et al., 2020); (8) **Small-loss Selection** (Gui et al., 2021); and (9) **Random Selection**. See Appendix C.3 for details.

3.2 Experimental Results

How does OLR perform under different noise ratios? The results in Figure 4 (see Table 6 in the Appendix) show that OLR consistently improves post-training performance across a wide range of noise ratios under both inactive and active noise settings. Under inactive noise, OLR achieves average gains of **3.6**% on six ID benchmarks and **3.3**% on three OOD benchmarks. Remarkably, even at a 50% noise ratio, OLR maintains performance comparable to that at 10%, underscoring its robustness. By contrast, active noise causes a sharper performance decline as noise increases, with the model eventually collapsing below its initial performance. Nevertheless, in this more challenging

Table 2: Results under the Realistic Mixed-Difficulty Scenario.

Method	In-Distribution							Out-of-Distribution			
	AIME24	AIME25	AMC	MATH-500	Minerva	Olympiad	**Avg.**	ARC-c	GPQA*	MMLU-Pro	**Avg.**
Qwen3-4B-Base	9.6	4.2	34.2	52.6	19.5	28.4	24.7	35.8	14.1	33.3	27.7
Fully Supervised	19.2	20.4	55.6	84.2	43.4	52.0	45.8	91.0	35.9	61.3	62.7
Active Noisy Label											
$\rho = 0.2$	16.7	10.0	51.1	76.4	37.9	41.5	38.9	82.9	31.8	57.9	57.5
w/ OLR	20.0	17.1	52.3	84.0	40.1	49.3	**43.8**	82.3	32.3	59.7	**58.1**
$\rho = 0.5$	11.3	5.4	45.0	77.0	31.6	41.3	35.3	78.9	24.7	51.7	51.8
w/ OLR	11.3	10.8	46.2	77.8	37.9	41.7	**37.6**	85.2	31.8	55.3	**57.4**
Inactive Noisy Label											
$\rho = 0.2$	14.6	12.1	54.2	80.0	42.6	43.3	41.1	87.6	28.3	56.3	57.4
w/ OLR	20.0	19.6	55.0	84.2	41.1	49.3	**44.9**	88.8	28.8	61.5	**59.7**
$\rho = 0.5$	17.5	11.3	52.6	78.2	41.5	45.3	41.1	83.5	32.3	54.7	56.8
w/ OLR	17.9	18.3	53.8	84.2	41.5	46.1	**43.6**	87.5	34.3	56.6	**59.5**

Table 3: Results under active noisy labels appear within samples of varying difficulty.

Method	In-Distribution							Out-of-Distribution			
	AIME24	AIME25	AMC	MATH-500	Minerva	Olympiad	**Avg.**	ARC-c	GPQA*	MMLU-Pro	**Avg.**
Qwen3-4B-Base	9.6	4.2	34.2	52.6	19.5	28.4	24.8	35.8	14.1	33.3	27.7
Fully Supervised	13.3	10.0	50.2	76.8	40.1	43.3	38.9	87.9	26.8	56.4	57.0
Easy											
w/o OLR	10.0	9.2	46.2	75.4	34.6	42.4	36.3	76.2	28.3	51.5	52.0
w/ OLR	15.4	7.9	45.2	77.4	37.1	39.6	**37.1**	78.5	28.3	52.8	**53.2**
Moderate											
w/o OLR	11.7	6.3	47.3	79.8	37.9	41.5	37.4	84.4	26.8	54.8	55.3
w/ OLR	14.2	10.0	48.0	77.4	37.9	43.0	**38.4**	84.6	28.8	55.7	**56.4**
Hard											
w/o OLR	12.9	7.9	48.8	79.0	37.9	42.1	38.1	84.7	28.8	55.0	56.2
w/ OLR	15.0	9.2	47.7	78.0	39.7	43.3	**38.8**	85.1	31.3	56.1	**57.5**

scenario, OLR still delivers substantial improvements: **3.9**% on ID benchmarks and **4.6**% on OOD benchmarks. These findings highlight OLR's ability to enhance robustness under inactive noise and achieve significant gains in complex active noise environments.

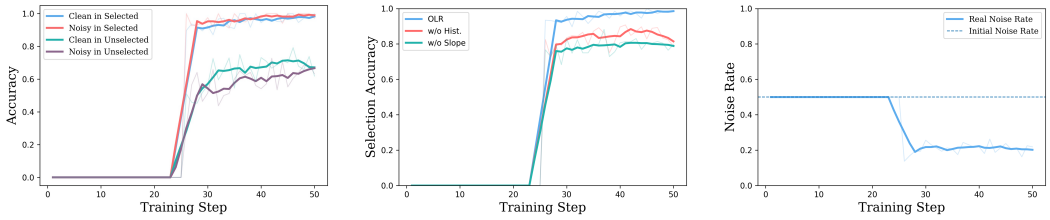
How does OLR perform compared to baseline methods? We present OLR's performance relative to baseline methods in Table 1 under a 50% noise-ratio scenario from two perspectives: unsupervised methods and traditional noisy label learning approaches. **(i) Unsupervised methods.** Most unsupervised methods lead to severe model collapse. The best among them, TTRL and Co-Reward, only marginally outperform naive noisy training on in-distribution tasks, suggesting that abandoning labels entirely is suboptimal. On out-of-distribution tasks, TTRL shows stronger robustness to active noise. Nevertheless, OLR achieves superior overall performance: average gains of **5.0**% and **8.0**% over unsupervised SOTA in ID settings under active and inactive noise, respectively, and outperforms the best unsupervised method by **3.3**% on OOD tasks. **(ii) Traditional noisy label learning methods.** We also find most noisy label methods ineffective in RLVR. Among active noisy label settings, only Confidence Penalty improves over naive training; others harm performance. Notably, small-loss selection declines most sharply, underperforming both baseline and random selection. This is because in RLVR, small loss frequently signals uninformative samples (e.g., rollouts entirely correct or incorrect) rather than high label quality, offering little improvement signal. In ID benchmarks, OLR outperforms the best of these methods by an average of **6.1**%.

How does OLR perform in scenarios with diverse sample difficulty? In practice, the difficulty of RLVR training samples varies, and the noisy label ratio is not extremely high. However, Table 2 shows that even a modest noise ratio such as 20% active noisy labels, which causes a 6.9% absolute performance drop, can still significantly harm model training, highlighting the non-negligible impact of label noise. At a 20% noise ratio, when active noisy labels are present, OLR improves ID tasks by 4.9% and OOD tasks by 0.6% on average, while with inactive noisy labels, it improves ID tasks by 3.8% and OOD tasks by 2.3% on average. At a 50% noise ratio, when active noisy labels are present, OLR improves ID tasks by 2.3% and OOD tasks by 5.6% on average, and with inactive noisy labels, it improves ID tasks by 2.5% and OOD tasks by 2.7% on average. These results demonstrate that OLR remains effective in mitigating label noise in scenarios with more diverse sample difficulty.

Does label noise affect easy and hard samples differently? We construct three difficulty-specific noisy-label datasets, corresponding to pass rates of 0.125 (hard), 0.500 (moderate), and 0.875 (easy). For each category, noisy labels are injected into one subset while keeping the others clean, enabling analysis of training under difficulty-targeted noise. The study of inactive noisy labels essentially explores how problem difficulty affects RLVR training (Qu et al., 2026; Zhan et al., 2025), the

Table 4: Training time and computational complexity of OLR under the High-Utilization Scenario.

Phase	Operation	Compute Complexity	Time
RLVR Training	Policy rollout and GRPO update	$\mathcal{O}(NKL)$	∼3.8 h
OLR Update	Majority estimation + statistics update	$\mathcal{O}(NK)$	negligible
Total	RLVR + OLR	$\mathcal{O}(NKL)$	∼3.8 h



(a) Selected vs. Unselected Acc. (b) Ablation Acc. (c) Real vs. Initial Noise

Figure 5: Training dynamics of OLR under the active noise setting ($\rho = 0.5$) and the High-Utilization Scenario. Acc. denotes accuracy.

interaction between active noisy labels and sample difficulty remains underexplored. To address this gap, we conduct experiments on stratified noise settings. Table 3 shows that noisy labels in easy samples cause the most severe performance drop, reducing ID to 36.3% and OOD to 52.0%, compared to 38.9% and 57.0% under clean labels. This is because corrupting easy samples undermines the model's foundational knowledge. In contrast, noise in hard samples leads to a smaller drop, suggesting the model is less dependent on those samples. Applying OLR robustly mitigates label noise across all difficulty levels, consistently improving the model's reasoning performance.

Does OLR help with naturally noisy reference labels? On BIRD Text2SQL, clean GRPO trains on corrected SQL, while noisy GRPO and OLR share the fixed original labels on the same inputs. Natural label noise lowers GRPO execution accuracy by 2.11–4.23 points across the three backbones. OLR recovers 57.3–87.5% of this drop (Table 5), with gains on Qwen3-8B and non-Qwen Olmo3-7B (Olmo et al., 2025). Appendix D.3 reports rollout feasibility and database-level examples.

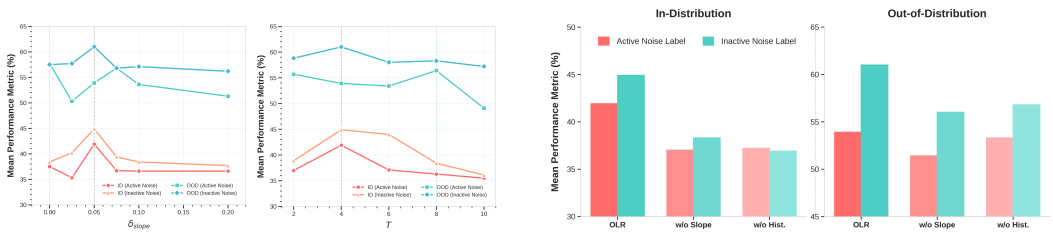
Table 5: BIRD Text2SQL test execution accuracy (%) with naturally noisy, fixed reference labels. Recovery is the fraction of the clean-to-noisy GRPO accuracy drop regained by OLR.

Method	Qwen3-4B	Qwen3-8B	Olmo3-7B
Clean GRPO	46.03	44.97	38.62
Noisy GRPO	41.80	42.86	34.92
OLR	**45.50**	**44.44**	**37.04**
Recovery	87.5%	75.0%	57.3%

How does OLR behave during the training process? Figure 5 illustrates training dynamics under 50% active noise. In **(a)**, for samples selected by OLR, the majority answer accuracy exceeds **90%** and approaches 100%, whereas unselected samples remain below 70%. Since unselected samples retain their original labels, the remaining noise consists only of original noisy labels (reduced by **30%**) and a small fraction (below 5%) of erroneous majority answers from selected samples. **(b)** demonstrates that combining both criteria achieves nearly 100% accuracy in sample selection, while removing either reduces accuracy by about 20%, highlighting the indispensability of both criteria. Finally, **(c)** demonstrates that OLR reduces the proportion of complex active noise by approximately **30%** during the refinement phase, thereby showcasing its robustness and effectiveness. Figure 10 illustrates the training dynamics in an inactive noise scenario.

Time Complexity Let N be the number of prompts, K the rollouts per prompt, and L the average rollout length. RLVR training is dominated by rollout generation, costing $O(NKL)$ per epoch. OLR adds majority-answer counting ($O(K)$ per prompt) and incremental statistics updates ($O(1)$), resulting in an extra $O(NK)$ per epoch. This overhead is negligible compared to $O(NKL)$. Empirical runtime (Table 4) confirms the analysis that there is negligible OLR overhead.

Parameter sensitivity analysis and ablation study. We examine OLR's sensitivity to the slope threshold δ_{slope} and early learning phase duration T (Figure 6a). Sweeping $\delta_{slope} \in \{0.0, 0.025, 0.05, 0.075, 0.1, 0.2\}$ shows optimal performance at 0.05 for ID and OOD under in-



(a) Sensitivity analysis (b) Ablation study

Figure 6: Results under the High-Utilization Scenario and 50% noise ratio.

active noise: too low introduces erroneous majority answers, too high yields insufficient refinement samples. Varying $T \in \{2, 4, 6, 8, 10\}$ yields optimal performance at $T = 4$ (counting from 0), where smaller T causes inaccurate slope estimation and larger T prolongs noisy training. Detailed results are reported in Tables 9 and 10. Ablation studies further show that removing either the slope or historical consistency criterion causes substantial performance drops under both active and inactive noise, across ID and OOD benchmarks (Figure 6b; Table 11). These findings confirm that the two criteria are complementary and jointly essential to OLR.

4 Related Work

Reinforcement Learning with Verifiable Rewards (RLVR) has proven highly effective for training reasoning models in domains with clear ground truth, such as math and code (Hu et al., 2025; Team et al., 2025; Mroueh, 2025; Li et al., 2025b). By replacing learned reward models with rule-based verifiers (Jaech et al., 2024; Guo et al., 2025), RLVR avoids the complexities of human preference modeling (Christiano et al., 2017; Ouyang et al., 2022), enabling stable training pipelines that have produced powerful models like DeepSeek-R1 (Guo et al., 2025) via algorithms such as GRPO (Shao et al., 2024; Liu et al., 2025; Yu et al., 2025; Zheng et al., 2025). Parallel work has investigated alternative unsupervised methods, including self-judgment (Wu et al., 2024; Yuan et al., 2024; Xiong et al., 2025), ensemble heads (Wang et al., 2024b; Zhou et al., 2025), and heuristics like entropy and majority voting (Agarwal et al., 2025; Li et al., 2025a; Zuo et al., 2025; Zhang et al., 2025a), to enable scalable online learning (Zhang et al., 2025a; Zhao et al., 2025). However, despite the growing body of RLVR research, the critical issue of learning under noisy labels remains largely unaddressed.

Noisy Label Learning is well-established, but its techniques rarely accommodate RLVR's generation and on-policy dynamics. Traditional methods fall into four categories. *Noise Transition Matrix Estimation* (Xiao et al., 2015; Chen & Gupta, 2015; Srivastava et al., 2014; Sukhbaatar et al., 2014; Collier et al., 2021; Bucarelli et al., 2023) learns flipping probabilities between labels, but is unsuitable for RLVR, where noisy labels form an open set without clear class boundaries. *Loss Correction* (Goodfellow et al., 2014; Pereyra et al., 2017; Zhang et al., 2017; Menon et al., 2020; Cheng et al., 2021) uses regularization to reduce overfitting to noise, but these classification-focused methods have limited applicability to RLVR's generative tasks. *Small-loss-based Sample Selection* (Malach & Shalev-Shwartz, 2017; Wang et al., 2018; Han et al., 2018b; Jiang et al., 2018; Wu et al., 2021; Song et al., 2021) treats low-loss samples as clean. However, in RLVR, small-loss samples are not necessarily beneficial (Zhan et al., 2025).

5 Conclusion

In this paper, we present a refined perspective on the impact of noisy labels in RLVR. First, we categorize noisy labels by rollout feasibility into active and inactive types, establishing a taxonomy to guide future research. Second, we identify Early Correctness Coherence, where clean and noisy samples improve similarly during early training before diverging later. Building on this insight, we propose Online Label Refinement (OLR), which dynamically denoises labels during RLVR training. Experiments demonstrate that OLR substantially enhances denoising effectiveness and improves reasoning performance across diverse noise ratios. Beyond RLVR, our approach opens promising avenues for broader applications, including VLM and agent domains, where noisy-label learning remains underexplored, thereby laying the groundwork for future advances across diverse paradigms.

AI USE STATEMENT

Generative AI assisted with code inspection, figure and table formatting, and improving the clarity of author-written text. The authors reviewed all AI-assisted changes and take full responsibility for the paper.

ETHICS STATEMENT

This work studies noisy reference labels in RLVR using mathematical reasoning benchmarks and BIRD Text2SQL. The reported results do not establish reliability in high-stakes applications, which require independent evaluation of labels, model behavior, and downstream risks.

REPRODUCIBILITY STATEMENT

Section 2 specifies the problem, assumptions, and OLR algorithm, with proofs in Appendix B. Appendix C.1–C.3 documents data construction, splits, training settings, and baselines; Appendix D reports additional controls and analyses. Code, configurations, and evaluation scripts will accompany the anonymized supplement.

REFERENCES

Shivam Agarwal, Zimin Zhang, Lifan Yuan, Jiawei Han, and Hao Peng. The unreasonable effectiveness of entropy minimization in llm reasoning. arXiv preprint arXiv:2505.15134, 2025.

Yingbin Bai, Erkun Yang, Bo Han, Yanhua Yang, Jiatong Li, Yinian Mao, Gang Niu, and Tongliang Liu. Understanding and improving early stopping for learning with noisy labels. Advances in Neural Information Processing Systems, 34:24392–24403, 2021.

David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. Advances in neural information processing systems, 32, 2019.

Maria Sofia Bucarelli, Lucas Cassano, Federico Siciliano, Amin Mantrach, and Fabrizio Silvestri. Leveraging inter-rater agreement for classification in the presence of noisy labels. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3439–3448, 2023.

Xin-Qiang Cai, Wei Wang, Feng Liu, Tongliang Liu, Gang Niu, and Masashi Sugiyama. Reinforcement learning with verifiable yet noisy rewards under imperfect verifiers. arXiv preprint arXiv:2510.00915, 2025.

Hao Chen, Ran Tao, Yue Fan, Yidong Wang, Jindong Wang, Bernt Schiele, Xing Xie, Bhiksha Raj, and Marios Savvides. Softmatch: Addressing the quantity-quality trade-off in semi-supervised learning. arXiv preprint arXiv:2301.10921, 2023.

Pengfei Chen, Ben Ben Liao, Guangyong Chen, and Shengyu Zhang. Understanding and utilizing deep neural networks trained with noisy labels. In International conference on machine learning, pp. 1062–1070. PMLR, 2019.

Xinlei Chen and Abhinav Gupta. Webly supervised learning of convolutional networks. In Proceedings of the IEEE international conference on computer vision, pp. 1431–1439, 2015.

Hao Cheng, Zhaowei Zhu, Xing Sun, and Yang Liu. Mitigating memorization of noisy labels via regularization between representations. arXiv preprint arXiv:2110.09022, 2021.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. Advances in neural information processing systems, 30, 2017.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. arXiv:1803.05457v1, 2018.

Mark Collier, Basil Mustafa, Efi Kokiopoulou, Rodolphe Jenatton, and Jesse Berent. Correlated input-dependent label noise in large-scale image classification. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 1551–1560, 2021.

Kazuki Egashira, Mark Vero, Jasper Dekoninck, Florian E Dorner, Robin Staab, and Martin Vechev. Delay, plateau, or collapse: Evaluating the impact of systematic verification error on rlvr. arXiv preprint arXiv:2605.02909, 2026.

Chuanwen Feng, Yilong Ren, and Xike Xie. Ot-filter: An optimal transport filter for learning with noisy labels. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 16164–16174, 2023.

Jacob Goldberger and Ehud Ben-Reuven. Training deep neural-networks using a noise adaptation layer. In International conference on learning representations, 2017.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572, 2014.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. arXiv preprint arXiv:2407.21783, 2024.

Xian-Jin Gui, Wei Wang, and Zhang-Hao Tian. Towards understanding deep learning from noisy labels with small-loss criterion. arXiv preprint arXiv:2106.09291, 2021.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. arXiv preprint arXiv:2501.12948, 2025.

Bo Han, Jiangchao Yao, Gang Niu, Mingyuan Zhou, Ivor Tsang, Ya Zhang, and Masashi Sugiyama. Masking: A new perspective of noisy supervision. Advances in neural information processing systems, 31, 2018a.

Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. Advances in neural information processing systems, 31, 2018b.

Bingxiang He, Yuxin Zuo, Zeyuan Liu, Shangziqi Zhao, Zixuan Fu, Junlin Yang, Cheng Qian, Kaiyan Zhang, Yuchen Fan, Ganqu Cui, et al. How far can unsupervised rlvr scale llm training? arXiv preprint arXiv:2603.08660, 2026.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, et al. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 3828–3850, 2024.

Dan Hendrycks, Mantas Mazeika, Duncan Wilson, and Kevin Gimpel. Using trusted data to train deep networks on labels corrupted by severe noise. Advances in neural information processing systems, 31, 2018.

Dan Hendrycks, Kimin Lee, and Mantas Mazeika. Using pre-training can improve model robustness and uncertainty. In International conference on machine learning, pp. 2712–2721. PMLR, 2019.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. arXiv preprint arXiv:2103.03874, 2021.

Jingcheng Hu, Yinmin Zhang, Qi Han, Daxin Jiang, Xiangyu Zhang, and Heung-Yeung Shum. Open-reasoner-zero: An open source approach to scaling up reinforcement learning on the base model, 2025. URL https://arxiv.org/abs/2503.24290.

Jinchi Huang, Lie Qu, Rongfei Jia, and Binqiang Zhao. O2u-net: A simple noisy label detection approach for deep neural networks. In Proceedings of the IEEE/CVF international conference on computer vision, pp. 3326–3334, 2019.

Yuzhen Huang, Weihao Zeng, Xingshan Zeng, Qi Zhu, and Junxian He. From accuracy to robustness: A study of rule-and model-based verifiers in mathematical reasoning. arXiv preprint arXiv:2505.22203, 2025.

Zhizhong Huang, Junping Zhang, and Hongming Shan. Twin contrastive learning with noisy labels. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 11661–11670, 2023.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. arXiv preprint arXiv:2412.16720, 2024.

Simon Jenni and Paolo Favaro. Deep bilevel learning. In Proceedings of the European conference on computer vision (ECCV), pp. 618–633, 2018.

Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In International conference on machine learning, pp. 2304–2313. PMLR, 2018.

Nazmul Karim, Mamshad Nayeem Rizve, Nazanin Rahnavard, Ajmal Mian, and Mubarak Shah. Unicon: Combating label noise through uniform selection and contrastive learning. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 9676–9686, 2022.

Taehyeon Kim, Jongwoo Ko, JinHwan Choi, Se-Young Yun, et al. Fine samples for learning with noisy labels. Advances in Neural Information Processing Systems, 34:24137–24149, 2021a.

Youngdong Kim, Juseung Yun, Hyounguk Shon, and Junmo Kim. Joint negative and positive learning for noisy labels. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 9442–9451, 2021b.

Kimin Lee, Sukmin Yun, Kibok Lee, Honglak Lee, Bo Li, and Jinwoo Shin. Robust inference via generative classifiers for handling noisy labels. In International conference on machine learning, pp. 3763–3772. PMLR, 2019.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, et al. Solving quantitative reasoning problems with language models. Advances in Neural Information Processing Systems, 35:3843–3857, 2022.

Jia Li, Edward Beeching, Lewis Tunstall, Ben Lipkin, Roman Soletskyi, Shengyi Huang, Kashif Rasul, Longhui Yu, Albert Q. Jiang, Ziju Shen, et al. Numinamath: The largest public dataset in ai4maths with 860k pairs of competition math problems and solutions. https://huggingface.co/datasets/Numinamath, 2024. Hugging Face repository, 13:9.

Junnan Li, Richard Socher, and Steven CH Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. arXiv preprint arXiv:2002.07394, 2020.

Pengyi Li, Matvey Skripkin, Alexander Zubrey, Andrey Kuznetsov, and Ivan Oseledets. Confidence is all you need: Few-shot rl fine-tuning of language models. arXiv preprint arXiv:2506.06395, 2025a.

Shikun Li, Xiaobo Xia, Shiming Ge, and Tongliang Liu. Selective-supervised contrastive learning with noisy labels. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 316–325, 2022.

Xuefeng Li, Haoyang Zou, and Pengfei Liu. Limr: Less is more for rl scaling. arXiv preprint arXiv:2502.11886, 2025b.

Yifan Li, Hu Han, Shiguang Shan, and Xilin Chen. Disc: Learning from noisy labels via dynamic instance-specific selection and correction. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 24070–24079, 2023.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective. arXiv preprint arXiv:2503.20783, 2025.

Michal Lukasik, Srinadh Bhojanapalli, Aditya Menon, and Sanjiv Kumar. Does label smoothing mitigate label noise? In International Conference on Machine Learning, pp. 6448–6458. PMLR, 2020.

Ang Lv, Ruobing Xie, Xingwu Sun, Zhanhui Kang, and Rui Yan. The climb carves wisdom deeper than the summit: On the noisy rewards in learning to reason. arXiv preprint arXiv:2505.22653, 2025.

Eran Malach and Shai Shalev-Shwartz. Decoupling" when to update" from" how to update". Advances in neural information processing systems, 30, 2017.

Aditya Krishna Menon, Ankit Singh Rawat, Sashank J Reddi, and Sanjiv Kumar. Can gradient clipping mitigate label noise? In International conference on learning representations, 2020.

Youssef Mroueh. Reinforcement learning with verifiable rewards: Grpo's effective loss, dynamics, and success amplification. arXiv preprint arXiv:2503.06639, 2025.

Duc Tam Nguyen, Chaithanya Kumar Mummadi, Thi Phuong Nhung Ngo, Thi Hoai Phuong Nguyen, Laura Beggel, and Thomas Brox. Self: Learning to filter noisy labels with self-ensembling. arXiv preprint arXiv:1910.01842, 2019.

Team Olmo, Allyson Ettinger, Amanda Bertsch, Bailey Kuehl, David Graham, David Heineman, Dirk Groeneveld, Faeze Brahman, Finbarr Timbers, Hamish Ivison, et al. Olmo 3. arXiv preprint arXiv:2512.13961, 2025.

Diego Ortego, Eric Arazo, Paul Albert, Noel E O'Connor, and Kevin McGuinness. Multi-objective interpolation training for robustness to label noise. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 6606–6615, 2021.

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.

Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: A loss correction approach. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1944–1952, 2017.

Gabriel Pereyra, George Tucker, Jan Chorowski, Łukasz Kaiser, and Geoffrey Hinton. Regularizing neural networks by penalizing confident output distributions. arXiv preprint arXiv:1701.06548, 2017.

Yun Qu, Qi Wang, Yixiu Mao, Vincent Tao Hu, Björn Ommer, and Xiangyang Ji. Can prompt difficulty be online predicted for accelerating rl finetuning of reasoning models? In Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1, pp. 1240–1250, 2026.

Yuntao Qu, Shasha Mo, and Jianwei Niu. Dat: Training deep networks robust to label-noise by matching the feature distributions. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 6821–6829, 2021.

Ali Rad, Khashayar Filom, Darioush Keivan, Peyman Mohajerin Esfahani, and Ehsan Kamalinejad. Rate or fate? rlv$^{\epsilon}$r: Reinforcement learning with verifiable noisy rewards. arXiv preprint arXiv:2601.04411, 2026.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level google-proof q&a benchmark. In First Conference on Language Modeling, 2024. URL https://openreview.net/forum?id=Ti67584b98.

Rulin Shao, Shuyue Stella Li, Rui Xin, Scott Geng, Yiping Wang, Sewoong Oh, Simon Shaolei Du, Nathan Lambert, Sewon Min, Ranjay Krishna, et al. Spurious rewards: Rethinking training signals in rlvr. arXiv preprint arXiv:2506.10947, 2025.

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.

Yanyao Shen and Sujay Sanghavi. Learning with bad training data via iterative trimmed loss minimization. In International conference on machine learning, pp. 5739–5748. PMLR, 2019.

Hwanjun Song, Minseok Kim, and Jae-Gil Lee. Selfie: Refurbishing unclean samples for robust deep learning. In International conference on machine learning, pp. 5907–5915. PMLR, 2019.

Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Robust learning by self-transition for handling noisy labels. In Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining, pp. 1490–1500, 2021.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research, 15(1):1929–1958, 2014.

Sainbayar Sukhbaatar, Joan Bruna, Manohar Paluri, Lubomir Bourdev, and Rob Fergus. Training convolutional networks with noisy labels. arXiv preprint arXiv:1406.2080, 2014.

Ryutaro Tanno, Ardavan Saeedi, Swami Sankaranarayanan, Daniel C Alexander, and Nathan Silberman. Learning from noisy labels by regularized estimation of annotator confusion. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 11244–11253, 2019.

Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, et al. Kimi k1. 5: Scaling reinforcement learning with llms. arXiv preprint arXiv:2501.12599, 2025.

Yikai Wang, Xinwei Sun, and Yanwei Fu. Scalable penalized regression for noise detection in learning with noisy labels. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 346–355, 2022.

Yisen Wang, Weiyang Liu, Xingjun Ma, James Bailey, Hongyuan Zha, Le Song, and Shu-Tao Xia. Iterative learning with open-set noisy labels. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 8688–8696, 2018.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. arXiv preprint arXiv:2406.01574, 2024a.

Zhaoyang Wang, Weilei He, Zhiyuan Liang, Xuchao Zhang, Chetan Bansal, Ying Wei, Weitong Zhang, and Huaxiu Yao. Cream: Consistency regularized self-rewarding language models. arXiv preprint arXiv:2410.12735, 2024b.

Hongxin Wei, Lei Feng, Xiangyu Chen, and Bo An. Combating noisy labels by agreement: A joint training method with co-regularization. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 13726–13735, 2020.

Hongxin Wei, Lue Tao, Renchunzi Xie, and Bo An. Open-set label noise can improve robustness against inherent label noise. Advances in Neural Information Processing Systems, 34:7978–7992, 2021.

Pengxiang Wu, Songzhu Zheng, Mayank Goswami, Dimitris Metaxas, and Chao Chen. A topological filter for learning with label noise. Advances in neural information processing systems, 33:21382–21393, 2020.

Tianhao Wu, Weizhe Yuan, Olga Golovneva, Jing Xu, Yuandong Tian, Jiantao Jiao, Jason Weston, and Sainbayar Sukhbaatar. Meta-rewarding language models: Self-improving alignment with llm-as-a-meta-judge. arXiv preprint arXiv:2407.19594, 2024.

Zhi-Fan Wu, Tong Wei, Jianwen Jiang, Chaojie Mao, Mingqian Tang, and Yu-Feng Li. Ngc: A unified framework for learning with open-world noisy data. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 62–71, 2021.

Xiaobo Xia, Tongliang Liu, Bo Han, Chen Gong, Nannan Wang, Zongyuan Ge, and Yi Chang. Robust early-learning: Hindering the memorization of noisy labels. In International conference on learning representations, 2020.

Tong Xiao, Tian Xia, Yi Yang, Chang Huang, and Xiaogang Wang. Learning from massive noisy labeled data for image classification. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2691–2699, 2015.

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

Jianhao Yan, Yafu Li, Zican Hu, Zhi Wang, Ganqu Cui, Xiaoye Qu, Yu Cheng, and Yue Zhang. Learning to reason under off-policy guidance. arXiv preprint arXiv:2504.14945, 2025.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. arXiv preprint arXiv:2505.09388, 2025a.

Shenzhi Yang, Guangcheng Zhu, Xing Zheng, Yingfan MA, Zhongqi Chen, Bowen Song, Weiqiang Wang, Junbo Zhao, Gang Chen, and Haobo Wang. Trapo: A semi-supervised reinforcement learning framework for boosting llm reasoning. arXiv preprint arXiv:2512.13106, 2025b.

Jiangchao Yao, Jiajie Wang, Ivor W Tsang, Ya Zhang, Jun Sun, Chengqi Zhang, and Rui Zhang. Deep learning from noisy image labels with quality embedding. IEEE Transactions on Image Processing, 28(4):1909–1922, 2018.

Yazhou Yao, Zeren Sun, Chuanyi Zhang, Fumin Shen, Qi Wu, Jian Zhang, and Zhenmin Tang. Jo-src: A contrastive approach for combating noisy labels. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 5192–5201, 2021.

William J Youden. Index for rating diagnostic tests. Cancer, 3(1):32–35, 1950.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. arXiv preprint arXiv:2503.14476, 2025.

Xingrui Yu, Bo Han, Jiangchao Yao, Gang Niu, Ivor Tsang, and Masashi Sugiyama. How does disagreement help generalization against label corruption? In International conference on machine learning, pp. 7164–7173. PMLR, 2019.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. Self-rewarding language models. arXiv preprint arXiv:2401.10020, 3, 2024.

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in llms beyond the base model? arXiv preprint arXiv:2504.13837, 2025.

Runzhe Zhan, Yafu Li, Zhi Wang, Xiaoye Qu, Dongrui Liu, Jing Shao, Derek F Wong, and Yu Cheng. Exgrpo: Learning to reason from experience. arXiv preprint arXiv:2510.02245, 2025.

Chang-Bin Zhang, Peng-Tao Jiang, Qibin Hou, Yunchao Wei, Qi Han, Zhen Li, and Ming-Ming Cheng. Delving deep into label smoothing. IEEE Transactions on Image Processing, 30:5984–5996, 2021a.

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. arXiv preprint arXiv:1710.09412, 2017.

Qingyang Zhang, Haitao Wu, Changqing Zhang, Peilin Zhao, and Yatao Bian. Right question is already half the answer: Fully unsupervised llm reasoning incentivization. arXiv preprint arXiv:2504.05812, 2025a.

Qingyang Zhang, Xinke Kong, Haitao Wu, Qinghua Hu, Minghao Wu, Baosong Yang, Yu Cheng, Yun Luo, Ganqu Cui, and Changqing Zhang. Tempo: Scaling test-time training for large reasoning models. arXiv preprint arXiv:2604.19295, 2026.

Yikai Zhang, Songzhu Zheng, Pengxiang Wu, Mayank Goswami, and Chao Chen. Learning with feature-dependent label noise: A progressive approach. arXiv preprint arXiv:2103.07756, 2021b.

Zizhuo Zhang, Jianing Zhu, Xinmu Ge, Zihua Zhao, Zhanke Zhou, Xuan Li, Xiao Feng, Jiangchao Yao, and Bo Han. Co-reward: Self-supervised reinforcement learning for large language model reasoning via contrastive agreement. arXiv preprint arXiv:2508.00410, 2025b.

Xuandong Zhao, Zhewei Kang, Aosong Feng, Sergey Levine, and Dawn Song. Learning to reason without external rewards. arXiv preprint arXiv:2505.19590, 2025.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, et al. Group sequence policy optimization. arXiv preprint arXiv:2507.18071, 2025.

Tianyi Zhou, Shengjie Wang, and Jeff Bilmes. Robust curriculum learning: from clean label detection to noisy label self-correction. In International conference on learning representations, 2020.

Xin Zhou, Yiwen Guo, Ruotian Ma, Tao Gui, Qi Zhang, and Xuanjing Huang. Self-consistency of the internal reward models improves self-rewarding language models. arXiv preprint arXiv:2502.08922, 2025.

Yuxuan Zhu and Daniel Kang. Noisy data is destructive to reinforcement learning with verifiable rewards. arXiv preprint arXiv:2603.16140, 2026.

Yuxin Zuo, Kaiyan Zhang, Shang Qu, Li Sheng, Xuekai Zhu, Biqing Qi, Youbang Sun, Ganqu Cui, Ning Ding, and Bowen Zhou. Ttrl: Test-time reinforcement learning. arXiv preprint arXiv:2504.16084, 2025.

A Additional Related Work

Semi-supervised Learning-based Methods (Berthelot et al., 2019; Zhou et al., 2020; Li et al., 2020; Kim et al., 2021a; Wang et al., 2022; Li et al., 2022; 2023; Feng et al., 2023): while abundant in classification, traditional semi-supervised methods are task-specific, which prevents direct transfer. Although recent work has explored semi-supervised RLVR (Yang et al., 2025b; Zhang et al., 2026), pre-selecting clean samples from noisy data remains an unsolved challenge.

RLVR under Noise primarily focus on *noisy rewards*. Rad et al. (2026) links GRPO success to the verifier's Youden index (Youden, 1950), noting that noise affects convergence. Egashira et al. (2026) distinguishes systematic errors from random noise, showing aggregate error rates are insufficient for reliability prediction. Huang et al. (2025) highlights the accuracy-robustness trade-off: rule-based verifiers suffer low recall, while model-based ones are prone to reward hacking. Cai et al. (2025) models verifier errors as stochastic noise and proposes policy gradient corrections for asymmetric error rates. Lv et al. (2025) introduces Reasoning Pattern Rewards (RPR) to leverage LLMs' inherent robustness to reward noise, boosting open-domain performance. In contrast, our work focuses on the more conventional label space of *noisy labels* in the training set. While Shao et al. (2025) attributes performance gains in Qwen-Math to amplified pretraining priors rather than genuine learning, and Zhu & Kang (2026) argues that data quality remains a critical bottleneck, with noise causing substantial degradation, *neither offers mitigation strategies*. Inspired by Early Correctness Coherence, we propose an online label-refinement method that significantly improves training effectiveness in noisy-label scenarios.

B Theoretical Proof

Theorem B.1 (Early Correctness Coherence in Noisy RLVR). *(Formal) Consider a dataset* $\mathcal{D} = \mathcal{D}_{\text{clean}} \cup \mathcal{D}_{\text{noise}}$ *with noise ratio* $\rho = |\mathcal{D}_{\text{noise}}|/|\mathcal{D}|$. *For each prompt* x, *let the policy have rollout probability* $p_t(y|x) = \pi_{\theta_t}(y|x)$ *with* K *rollouts per step. Assume small learning rate* η *and early-phase dynamics (ignoring clipping effects). Suppose (i) initial rollout bias* $p_0(y^\star) > p_0(\tilde{y})$ *creating a signal gap, (ii) positive cross-sample coupling*

$$\Gamma(x_c, x_n) = \nabla_\theta \log \pi(y^\star(x_c)|x_c) \cdot \nabla_\theta \log \pi(y^\star(x_n)|x_n), \\ \mathbb{E}_{x_c \in \mathcal{D}_{\text{clean}},\, x_n \in \mathcal{D}_{\text{noise}}}[\Gamma(x_c, x_n)] \geq \gamma > 0, \tag{8}$$

so that reinforcing clean samples increases correct probability on noisy samples, and (iii) average deterministic drift of the log-ratio $\Delta_s := \gamma(1-\rho)G_c - \rho G_n$, where G_c and G_n are the mean clean and noisy advantage magnitudes. Define the log-ratio

$$L_t(x) = \log \frac{p_t(y^\star(x)|x)}{p_t(\tilde{y}(x)|x)} \quad \textit{and} \quad \rho_c = \frac{\gamma G_c}{\gamma G_c + G_n}. \tag{9}$$

If $\rho < \rho_c$ and $K \gtrsim \log(T/\delta)$, then with probability at least $1-\delta$ simultaneously for all $t \le T$, where T denotes the early convergence stage,

$$L_t(x) \ge L_0(x) + \eta t\Big(\Delta_s - O\big(\sqrt{\frac{\log(T/\delta)}{K}}\big)\Big), \tag{10}$$

implying that $p_t(y^\star(x)|x) \gg p_t(\tilde{y}(x)|x)$.

B.1 Notation and Setup

We consider a dataset $\mathcal{D} = \mathcal{D}_{\text{clean}} \cup \mathcal{D}_{\text{noise}}$ with noise ratio

$$\rho = \frac{|\mathcal{D}_{\text{noise}}|}{|\mathcal{D}|}.$$

For each prompt $x \in \mathcal{D}$, let $y^\star(x)$ denote the correct solution and $\tilde{y}(x)$ a wrong reference label. The log-ratio analysis below assumes positive policy probabilities for both answers; this support assumption is distinct from their observed finite-K rollout status. The policy is $\pi_\theta(y|x)$, and we define

$$p_t(y|x) = \pi_{\theta_t}(y|x),$$

with K rollouts per step. Binary reward:

$$r(x,y) \in \{0,1\}, \quad A(x,y) = \frac{r(x,y) - \mu(x)}{\sigma(x) + \epsilon}.$$

B.2 Finite-Rollout Concentration Bounds

Lemma B.2 (Advantage Concentration)**.** *For any prompt x, let $\hat{\mu}(x)$ and $\hat{\sigma}(x)$ be the empirical mean and standard deviation computed from K rollouts. Then, for any $\delta > 0$, with probability at least $1-\delta$:*

$$|\hat{\mu}(x) - \mathbb{E}[\mu(x)]| \le \sqrt{\frac{\log(2/\delta)}{2K}}, \tag{11}$$

$$|\hat{\sigma}(x) - \sigma(x)| \le C\sqrt{\frac{\log(2/\delta)}{K}}, \tag{12}$$

$$|\hat{A}(x,y) - A(x,y)| \le C'\sqrt{\frac{\log(1/\delta)}{K}}, \tag{13}$$

where C, C' are constants depending on reward bounds.

Proof. We start with the empirical mean of K rollouts for a fixed prompt x:

$$\hat{\mu}(x) = \frac{1}{K}\sum_{k=1}^{K} r(x, y^{(k)}),$$

where each $r(x, y^{(k)}) \in [0,1]$ and $y^{(k)} \sim \pi_\theta(\cdot|x)$.

Step 1: Concentration of the empirical mean. Since the $r(x, y^{(k)})$ are independent conditioned on the policy, we can apply Hoeffding's inequality:

$$\Pr\Big(|\hat{\mu}(x) - \mathbb{E}[\hat{\mu}(x)]| > \epsilon\Big) \le 2\exp\Big(-2K\epsilon^2\Big),$$

where $\mathbb{E}[\hat{\mu}(x)] = \mu(x)$ is the true expected reward for this prompt. Solving for ϵ with confidence level $1-\delta$ gives:

$$|\hat{\mu}(x) - \mu(x)| \le \sqrt{\frac{\log(2/\delta)}{2K}} \quad \text{with probability at least } 1-\delta. \tag{14}$$

Step 2: Concentration of the empirical variance. The empirical variance is

$$\hat{\sigma}^2(x) = \frac{1}{K}\sum_{k=1}^{K}(r(x, y^{(k)}) - \hat{\mu}(x))^2.$$

Since $0 \le r(x, y^{(k)}) \le 1$, we have bounded deviations:

$$|(r(x, y^{(k)}) - \hat{\mu}(x))^2 - (r(x, y^{(k)}) - \mu(x))^2| \le |\hat{\mu}(x) - \mu(x)|. \tag{15}$$

Thus, the empirical variance is a 1-Lipschitz function of the empirical mean. Combining this with Hoeffding's inequality for bounded r gives

$$|\hat{\sigma}(x) - \sigma(x)| \le C\sqrt{\frac{\log(2/\delta)}{K}}$$

for some constant C.

Step 3: Concentration of normalized advantage. The normalized advantage is

$$\hat{A}(x, y) = \frac{r(x, y) - \hat{\mu}(x)}{\hat{\sigma}(x) + \epsilon}.$$

Consider the mapping $(r, \hat{\mu}, \hat{\sigma}) \mapsto \hat{A}(x, y)$. This mapping is Lipschitz in each argument: a deviation δr, $\delta\hat{\mu}$, $\delta\hat{\sigma}$ induces a change

$$|\delta\hat{A}| \le \frac{|\delta r| + |\delta\hat{\mu}|}{\sigma + \epsilon} + \frac{|r - \mu|}{(\sigma + \epsilon)^2}|\delta\hat{\sigma}|.$$

Since r, μ, σ are bounded in $[0, 1]$, the total Lipschitz constant is finite. Therefore, combining the bounds from Steps 1–2 and applying union bound over numerator and denominator terms, we obtain

$$|\hat{A}(x, y) - A(x, y)| \le C'\sqrt{\frac{\log(1/\delta)}{K}},$$

with probability at least $1 - \delta$, for some constant C' depending on the reward bounds and ϵ.

□

B.3 Log-Ratio Dynamics and Martingale Decomposition

Define the log-ratio for a active-noise sample x_n:

$$L_t(x_n) = \log\frac{p_t(y^\star(x_n)|x_n)}{p_t(\tilde{y}(x_n)|x_n)}.$$

Lemma B.3 (Finite-Rollout Log-Ratio Dynamics with High-Probability Bound)**.** *Let $L_t(x_n) = \log\frac{p_t(y^\star(x_n)|x_n)}{p_t(\tilde{y}(x_n)|x_n)}$ be the log-ratio of clean vs active-noise solutions for sample x_n, and let the one-step update satisfy*

$$\Delta L_t(x_n) = \eta\big(\hat{A}_t(x_n, y^\star) - \hat{A}_t(x_n, \tilde{y})\big) + \eta\Xi_t(x_n),$$

where $\Xi_t(x_n)$ is the cross-sample coupling contribution. Define the martingale difference

$$M_t(x_n) = \Delta L_t(x_n) - \mathbb{E}[\Delta L_t(x_n) \mid \mathcal{F}_{t-1}].$$

Assume the finite-rollout advantage satisfies Lemma B.2. Then for rollout size K sufficiently large and learning rate η small, with probability at least $1 - \delta$,

$$\left|L_t(x_n) - L_0(x_n) - \eta\sum_{s=0}^{t-1}\mathbb{E}[\Delta L_s(x_n) \mid \mathcal{F}_{s-1}]\right| \le t\eta C\sqrt{\frac{\log(1/\delta)}{K}} \quad \text{for all } t \le T, \tag{16}$$

where $C > 0$ is a constant depending on reward bounds. In particular, $\{M_t(x_n), \mathcal{F}_t\}$ is a martingale difference sequence with bounded increments.

Proof. Recall the log-ratio for a active-noise sample x_n:

$$L_t(x_n) = \log\frac{p_t(y^\star(x_n)|x_n)}{p_t(\tilde{y}(x_n)|x_n)}.$$

Step 1: One-step stochastic update. Under GRPO, the parameter update is

$$\theta_{t+1} = \theta_t + \eta \sum_{y \in \mathcal{Y}_t(x_n)} \hat{A}_t(x_n, y) \nabla_\theta \log \pi_\theta(y|x_n),$$

where $\hat{A}_t$ is computed from K rollouts. Using first-order approximation for small η, the log-ratio change is

$$\Delta L_t(x_n) := L_{t+1}(x_n) - L_t(x_n) \approx \eta\Big(\hat{A}_t(x_n, y^\star) - \hat{A}_t(x_n, \tilde{y})\Big) + \eta \Xi_t(x_n), \tag{17}$$

where Ξ_t is the cross-sample coupling term arising from shared parameters across prompts.

Step 2: Conditional expectation and martingale decomposition. Define the filtration $\mathcal{F}_{t-1}$ as the sigma-algebra generated by all previous rollouts and policy parameters up to step $t-1$. Conditional on $\mathcal{F}_{t-1}$, the stochastic gradient $\hat{A}_t(x_n, y)$ is unbiased:

$$\mathbb{E}[\hat{A}_t(x_n, y) \mid \mathcal{F}_{t-1}] = A_t(x_n, y),$$

since the rollouts $y^{(k)} \sim \pi_{\theta_t}(\cdot|x_n)$ are independent given θ_t.

Define the martingale difference sequence:

$$M_t(x_n) := \Delta L_t(x_n) - \mathbb{E}[\Delta L_t(x_n) \mid \mathcal{F}_{t-1}].$$

By construction:

1. $\mathbb{E}[M_t(x_n) \mid \mathcal{F}_{t-1}] = 0$, i.e., $\{M_t(x_n), \mathcal{F}_t\}$ is a martingale difference sequence. 2. The magnitude is bounded due to bounded rewards and Lemma B.2:

$$|M_t(x_n)| = |\Delta L_t(x_n) - \mathbb{E}[\Delta L_t(x_n) \mid \mathcal{F}_{t-1}]| \le \eta C \sqrt{\frac{\log(1/\delta)}{K}} \quad \text{with probability } 1-\delta, \tag{18}$$

where the constant C comes from the Lipschitz dependence of the log-ratio on normalized advantage.

Step 3: Martingale decomposition. With this definition, we can write

$$L_{t+1}(x_n) = L_0(x_n) + \eta \sum_{s=0}^{t} \Delta_s(x_n) + \sum_{s=0}^{t} M_s(x_n),$$

where $\Delta_s(x_n) = \mathbb{E}[\Delta L_s(x_n) \mid \mathcal{F}_{s-1}]$ is the deterministic drift term. This decomposition separates the deterministic drift from the stochastic fluctuation (martingale), allowing application of Azuma-Hoeffding inequality to bound deviations of $L_t(x_n)$ around its mean trajectory.

Step 4: High-probability bound. Applying Azuma-Hoeffding to $\sum_{s=0}^{t} M_s(x_n)$, since each $|M_s(x_n)| \le \eta C \sqrt{\log(1/\delta)/K}$, we obtain

$$\Pr\Big(\Big|\sum_{s=0}^{t} M_s(x_n)\Big| \ge \epsilon\Big) \le 2\exp\Big(-\frac{\epsilon^2}{2t\eta^2 C^2 \log(1/\delta)/K}\Big). \tag{19}$$

Setting $\epsilon = t\eta C\sqrt{\log(1/\delta)/K}$ gives probability at most δ. □

B.4 Cross-Sample Coupling

Define cross-sample coupling:

$$\Gamma(x_i, x_j) = \nabla_\theta \log \pi(y^\star(x_j)|x_j) \cdot \nabla_\theta \log \pi(y^\star(x_i)|x_i). \tag{20}$$

Assume:

$$\mathbb{E}_{x_c \sim \mathcal{D}_{\text{clean}}, x_n \sim \mathcal{D}_{\text{noise}}}[\Gamma(x_c, x_n)] \ge \gamma > 0.$$

Then the deterministic drift for noise sample correct log-probability is:

$$\Delta_t = \gamma(1-\rho)G_c - \rho G_n,$$

where G_c and G_n denote average advantage magnitudes over clean and noisy samples.

B.5 High-Probability Early Correctness Coherence

Theorem B.4 (High-Probability Early Correctness Coherence). *Suppose* $p_0(y^\star) > p_0(\tilde{y})$, $\gamma > 0$, *and* $K \gtrsim \log(T/\delta)$. *Define*

$$\rho_c = \frac{\gamma G_c}{G_n + \gamma G_c}.$$

If $\rho < \rho_c$, *then with probability at least* $1 - \delta$,

$$L_t(x_n) \geq L_0(x_n) + \eta t\Big(\gamma(1-\rho)G_c - \rho G_n - O(\sqrt{\frac{\log(T/\delta)}{K}})\Big) \tag{21}$$

for all $t \leq T$. *In particular,* $L_t(x_n)$ *increases monotonically in early phase, implying*

$$p_t(y^\star(x_n)|x_n) \gg p_t(\tilde{y}(x_n)|x_n).$$

Proof. Sum over martingale differences:

$$L_t = L_0 + \eta \sum_{s<t} \Delta_s + \sum_{s<t} M_s.$$

By Azuma-Hoeffding inequality, with probability $1 - \delta$, $|\sum M_s| \leq t\eta C\sqrt{\log(T/\delta)/K}$. The deterministic drift term is positive under $\rho < \rho_c$, yielding the stated high-probability bound. □

B.6 Active Noise Collapse Condition

Theorem B.5 (Active-Noise Collapse). *If* $\rho > \rho_c + O(\sqrt{\log(T/\delta)/K})$, *then with high probability* $L_t(x_n) \to -\infty$, *i.e., active-noise solutions dominate.*

Proof. Recall the log-ratio for a active-noise sample x_n:

$$L_t(x_n) = \log \frac{p_t(y^\star(x_n)|x_n)}{p_t(\tilde{y}(x_n)|x_n)}.$$

Step 1: Martingale decomposition. From Lemma B.3, we have the decomposition

$$L_t(x_n) = L_0(x_n) + \eta \sum_{s=0}^{t-1} \Delta_s(x_n) + \sum_{s=0}^{t-1} M_s(x_n),$$

where $\Delta_s(x_n) = \mathbb{E}[\Delta L_s(x_n) \mid \mathcal{F}_{s-1}]$ is the deterministic drift and $\{M_s(x_n)\}$ is a martingale difference sequence with bounded increments

$$|M_s(x_n)| \leq \eta C\sqrt{\frac{\log(1/\delta)}{K}} \quad \text{w.p. } 1 - \delta.$$

Step 2: Drift under active-noise dominance. By definition, the deterministic drift is

$$\Delta_s(x_n) = \mathbb{E}[\Delta L_s(x_n) \mid \mathcal{F}_{s-1}] = \gamma(1-\rho)G_c - \rho G_n,$$

where G_c and G_n are the average clean and noisy advantage magnitudes, and $\gamma > 0$ is the cross-sample coupling coefficient.

Under the condition

$$\rho > \rho_c + O\Big(\sqrt{\frac{\log(T/\delta)}{K}}\Big), \quad \text{with } \rho_c = \frac{\gamma G_c}{G_n + \gamma G_c},$$

we have

$$\Delta_s(x_n) \leq -\epsilon < 0$$

for some positive ϵ depending on $(\rho - \rho_c)$ and finite-rollout corrections. This ensures that the deterministic drift is negative at every step.

Step 3: Finite-rollout martingale corrections. The martingale sum $\sum_{s=0}^{t-1} M_s(x_n)$ introduces stochastic fluctuations. By Azuma-Hoeffding inequality, for all $t \leq T$:

$$\Pr\left(\Big|\sum_{s=0}^{t-1} M_s(x_n)\Big| \geq t\eta C\sqrt{\frac{\log(1/\delta)}{K}}\right) \leq \delta.$$

For sufficiently large K, the martingale fluctuation is smaller in magnitude than the deterministic negative drift:

$$t\eta C\sqrt{\frac{\log(1/\delta)}{K}} < t\eta\epsilon,$$

so the stochastic perturbation does not reverse the sign of the drift.

Step 4: Log-ratio monotonic decrease. Combining Steps 2 and 3, we have with probability at least $1-\delta$:

$$L_t(x_n) = L_0(x_n) + \underbrace{\sum_{s=0}^{t-1} \Delta_s(x_n)}_{\text{negative drift}} + \underbrace{\sum_{s=0}^{t-1} M_s(x_n)}_{\text{bounded fluctuation}} \leq L_0(x_n) - t\eta(\epsilon - C\sqrt{\frac{\log(1/\delta)}{K}}). \tag{22}$$

As $t \to \infty$, the negative drift dominates and the log-ratio decreases monotonically, hence

$$L_t(x_n) \to -\infty,$$

implying that

$$p_t(y^\star(x_n)|x_n) \ll p_t(\tilde{y}(x_n)|x_n),$$

i.e., active-noise solutions dominate the policy with high probability.

□

B.7 KL-Regularized Dynamics

With KL regularization term $\beta D_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}})$, the replicator drift becomes

$$\Delta_t^{KL} = \gamma(1-\rho)G_c - \rho G_n - \beta\Delta_{\text{ref}},$$

where

$$\Delta_{\text{ref}} = \log\frac{p_{\text{ref}}(y^\star)}{p_{\text{ref}}(\tilde{y})}.$$

Theorem B.6 (KL-Regularized Phase Boundary)**.** *With KL regularization, the critical noise ratio shifts to*

$$\rho_c^{KL} = \frac{\gamma G_c - \beta\Delta_{ref}}{G_n + \gamma G_c}.$$

The high-probability Early Correctness Coherence and collapse conditions are modified accordingly.

Proof. We consider the same log-ratio for a active-noise sample x_n:

$$L_t(x_n) = \log\frac{p_t(y^\star(x_n)|x_n)}{p_t(\tilde{y}(x_n)|x_n)}.$$

Step 1: One-step update with KL regularization. Under GRPO with KL regularization, the surrogate gradient includes a term:

$$-\beta\nabla_\theta D_{\text{KL}}(\pi_\theta\|\pi_{\text{ref}}) = -\beta\mathbb{E}_{y\sim\pi_\theta}[\nabla_\theta \log\pi_\theta(y|x_n)\log\frac{\pi_\theta(y|x_n)}{\pi_{\text{ref}}(y|x_n)}]. \tag{23}$$

Using the same first-order approximation as before for small η, the deterministic drift of the log-ratio becomes

$$\Delta_t^{KL} := \mathbb{E}[\Delta L_t(x_n) \mid \mathcal{F}_{t-1}] = \gamma(1-\rho)G_c - \rho G_n - \beta\Delta_{\text{ref}}, \tag{24}$$

where

$$\Delta_{\text{ref}} = \log\frac{p_{\text{ref}}(y^\star)}{p_{\text{ref}}(\tilde{y})}$$

captures the bias introduced by the reference policy.

Step 2: Modified critical noise ratio. In the non-KL case, the phase boundary is

$$\rho_c = \frac{\gamma G_c}{\gamma G_c + G_n}.$$

Including the KL drift, we require the total deterministic drift to be zero at the boundary:

$$\Delta_t^{KL} = \gamma(1-\rho_c^{KL})G_c - \rho_c^{KL} G_n - \beta\Delta_{\text{ref}} = 0.$$

Solving for ρ_c^{KL} gives

$$\gamma(1-\rho_c^{KL})G_c - \rho_c^{KL} G_n - \beta\Delta_{\text{ref}} = 0 \quad \Rightarrow \quad \rho_c^{KL} = \frac{\gamma G_c - \beta\Delta_{\text{ref}}}{G_n + \gamma G_c}. \tag{25}$$

Step 3: Martingale decomposition and high-probability bound. Define the martingale difference as in Lemma B.3:

$$M_t(x_n) = \Delta L_t(x_n) - \mathbb{E}[\Delta L_t(x_n) \mid \mathcal{F}_{t-1}].$$

Its magnitude is still bounded by $\eta C\sqrt{\log(1/\delta)/K}$ due to Lemma B.2.

Thus, the log-ratio evolves as

$$L_t(x_n) = L_0(x_n) + \eta \sum_{s=0}^{t-1} \Delta_s^{KL} + \sum_{s=0}^{t-1} M_s(x_n),$$

and applying Azuma-Hoeffding gives with probability $1-\delta$:

$$\left| L_t(x_n) - L_0(x_n) - \eta \sum_{s=0}^{t-1} \Delta_s^{KL} \right| \le t\eta C\sqrt{\frac{\log(1/\delta)}{K}}. \tag{26}$$

Step 4: Implications for Early Correctness Coherence and collapse. - If $\rho < \rho_c^{KL}$, the deterministic drift $\Delta_t^{KL} > 0$, so $L_t(x_n)$ increases, and Early Correctness Coherence occurs. - If $\rho > \rho_c^{KL}$, the deterministic drift $\Delta_t^{KL} < 0$, so $L_t(x_n) \to -\infty$ with high probability, i.e., active-noise collapse occurs.

Finite-rollout stochastic fluctuations do not change the sign of the drift if K is sufficiently large, preserving the phase boundary.

This completes the proof. □

B.8 Conditional Noise-Rate Reduction under OLR

Proof of Theorem 2.4. Let $Z = n$ or c indicate whether a uniformly drawn prompt's original label is incorrect or correct. Write S for the event that OLR's slope and historical-consistency gates select a replacement, and E for the event that the current majority answer differs from the true answer. All probabilities below include the randomness of the training history and the current rollouts. Set

$$q_z = \Pr(S \mid Z = z), \qquad e_z = \Pr(E \mid S, Z = z), \quad z \in \{n, c\},$$

with $q_z e_z = 0$ when $q_z = 0$. The definition of e_z explicitly conditions on OLR's selection, which may use the same rollouts as the majority vote.

The majority-vote bound and selection. An exponential bound requires a probability gap against every competing answer class; a gap against the original wrong label alone is insufficient. For an optional sufficient condition, fix the history H before the current K rollouts and a prompt x. Suppose these rollouts are conditionally independent, their answers have at most M classes with positive probability, and uniformly over the histories and prompts under consideration,

$$\pi_{\theta(H)}(y^\star \mid x) - \max_{a \neq y^\star} \pi_{\theta(H)}(a \mid x) \ge g > 0.$$

For each competing class a, let $X_i = \mathbf{1}\{Y_i = y^\star\} - \mathbf{1}\{Y_i = a\} \in [-1, 1]$. Then $\mathbb{E}[X_i \mid H, x] \ge g$, so Hoeffding's inequality and a union bound give, for any tie-breaking rule,

$$\Pr(E \mid H, x) \le \Pr(\exists a \neq y^\star : N_a \ge N_{y^\star} \mid H, x) \le (M-1)e^{-Kg^2/2} =: \epsilon_K.$$

Table 6: Results on Qwen3-4B-Base under the High-Utilization Scenario with different noise ratios. Each numerical subscript denotes the absolute **improvement** or **degradation** compared with naive GRPO without OLR. **Bold** values indicate better performance.

Method	In-Distribution							Out-of-Distribution			
	AIME24	AIME25	AMC	MATH-500	Minerva	Olympiad	**Avg.**	ARC-c	GPQA*	MMLU-Pro	**Avg.**
Qwen3-4B-Base	9.6	4.2	34.2	52.6	19.5	28.4	24.8	35.8	14.1	33.3	27.7
Fully Supervised	20.4	17.9	52.9	85.0	42.6	48.1	44.5	86.9	36.4	59.1	60.8
Inactive Noisy Label											
$\rho = 0.1$	21.3	20.4	52.7	83.6	39.7	49.3	44.5	86.3	37.4	59.0	60.9
w/ OLR	22.5	19.6	55.0	83.0	41.2	48.3	**44.9**$_{+0.4}$	88.9	37.9	59.5	**62.1**$_{+1.2}$
$\rho = 0.3$	12.9	8.3	50.2	75.8	42.0	43.0	38.7	88.1	29.8	56.2	58.0
w/ OLR	23.8	16.7	53.5	82.6	40.8	48.7	**44.4**$_{+5.7}$	88.1	36.4	57.6	**60.7**$_{+2.7}$
$\rho = 0.5$	12.5	5.0	45.3	76.0	34.9	42.7	36.1	86.7	28.8	56.0	57.2
w/ OLR	21.7	18.3	53.9	84.2	42.3	48.9	**44.9**$_{+8.8}$	86.8	38.4	57.8	**61.0**$_{+3.8}$
$\rho = 0.7$	9.6	8.8	45.9	76.4	37.5	42.7	36.8	83.4	26.3	54.5	54.7
w/ OLR	14.6	11.3	48.0	79.2	37.1	43.4	**38.9**$_{+2.1}$	87.4	29.8	57.3	**58.2**$_{+3.5}$
$\rho = 0.9$	12.5	9.6	44.3	73.4	37.5	40.0	36.2	76.5	27.3	51.5	51.8
w/ OLR	10.8	9.6	48.2	76.6	37.1	42.2	**37.4**$_{+1.2}$	85.9	30.3	55.3	**57.2**$_{+5.4}$
Active Noisy Label											
$\rho = 0.1$	13.8	9.6	48.0	77.8	37.5	42.4	38.2	84.6	30.3	55.5	56.8
w/ OLR	19.2	19.2	52.0	81.6	39.7	52.0	**44.0**$_{+5.8}$	86.8	35.4	59.1	**60.4**$_{+3.6}$
$\rho = 0.3$	13.8	9.6	47.1	73.6	36.0	41.8	37.0	80.8	25.3	51.7	52.6
w/ OLR	13.8	12.9	50.3	77.4	36.4	41.3	**38.7**$_{+1.7}$	85.4	33.3	56.7	**58.5**$_{+5.9}$
$\rho = 0.5$	10.4	8.8	45.8	74.6	33.1	40.4	35.5	73.9	24.7	48.6	49.1
w/ OLR	20.4	15.4	49.7	81.4	36.4	48.1	**41.9**$_{+6.4}$	78.5	28.3	54.8	**53.9**$_{+4.8}$
$\rho = 0.7$	10.4	9.2	45.5	74.2	31.6	39.0	35.0	66.2	24.7	49.2	46.7
w/ OLR	10.8	10.0	47.1	76.2	37.5	41.5	**37.2**$_{+2.2}$	80.1	31.8	52.4	**54.8**$_{+8.1}$
$\rho = 0.9$	8.3	5.8	27.0	43.0	21.7	24.3	21.7	34.2	12.6	30.5	25.8
w/ OLR	7.1	5.8	35.7	47.8	25.7	28.6	**25.1**$_{+3.4}$	31.1	16.2	32.1	**26.5**$_{+0.7}$

This assumption is stronger than the pairwise log-ratio statement in Theorem B.1; we do not infer it from that theorem. Averaging over H and x yields $\Pr(E \mid Z = z) \le \epsilon_K$ if the gap holds in group z. Since $E \cap S \subseteq E$ even when S and E depend on the same rollout batch,

$$q_z e_z = \Pr(E \cap S \mid Z = z) \le \epsilon_K, \qquad e_z \le \min\{1, \epsilon_K / q_z\} \quad (q_z > 0).$$

There is no general bound $e_z \le \epsilon_K$ without an assumption controlling selection; the exact noise-rate identity below does not require this optional gap condition.

Effective noise rate. An originally incorrect label remains incorrect if it is not selected or its replacement is wrong, with conditional probability $1 - q_n + q_n e_n$. An originally correct label becomes incorrect only if it is selected and replaced incorrectly, with conditional probability $q_c e_c$. The law of total probability therefore gives

$$\rho_{\text{eff}} = \rho(1 - q_n + q_n e_n) + (1 - \rho) q_c e_c, \qquad \rho_{\text{eff}} - \rho = -\rho q_n (1 - e_n) + (1 - \rho) q_c e_c.$$

The claimed strict reduction holds exactly when $\rho q_n (1 - e_n) > (1 - \rho) q_c e_c$.

Conditional tolerance interpretation. Suppose separately that a learning criterion remains valid whenever the effective noise rate is below a baseline threshold $0 < \rho_c < 1$. If q_n, q_c, e_n, e_c are constant as ρ varies, put $a = 1 - q_n(1 - e_n)$ and $b = q_c e_c$. When $a > b$ and $b < \rho_c$, its boundary in terms of the original noise rate is

$$\rho_c^{\text{OLR}} = \frac{\rho_c - b}{a - b},$$

restricted to $[0, 1]$. It exceeds ρ_c precisely when $\rho_c q_n (1 - e_n) > (1 - \rho_c) q_c e_c$. If the rates vary with ρ, the condition is instead $\rho_{\text{eff}}(\rho) < \rho_c$; no universal factor $1/(1 - \Delta)$ follows. This transfer of a baseline threshold to corrected labels is an additional assumption, not a consequence of majority voting. □

C Experiment Details

C.1 Datasets and Splits

We use Qwen3-4B-Base to perform inference on the DAPO-Math dataset (Yu et al., 2025), sampling each example eight times and computing the pass rate to quantify problem difficulty, where lower

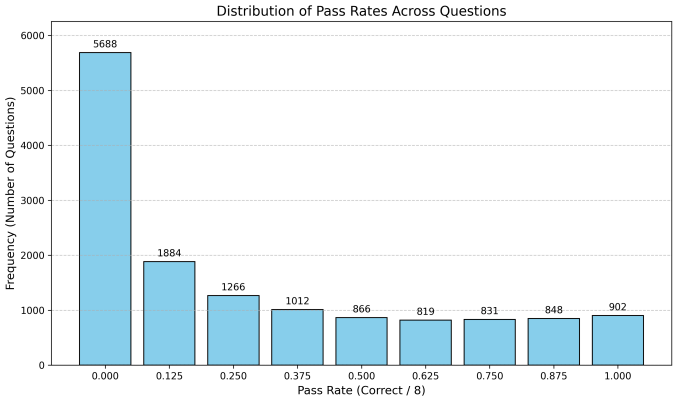


Figure 7: Difficulty distribution of the DAPO-Math dataset evaluated with Qwen3-4B-Base, measured by pass rate.

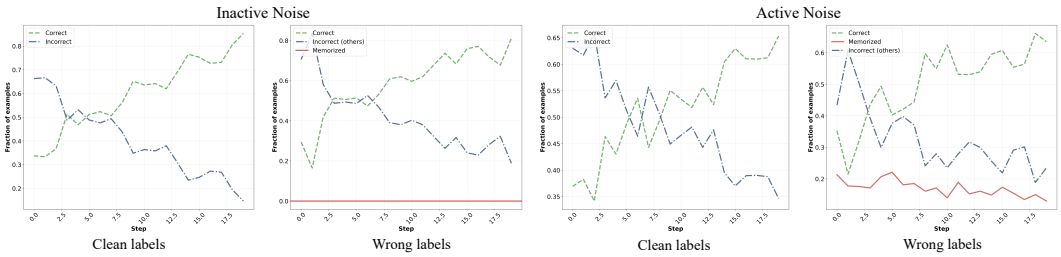
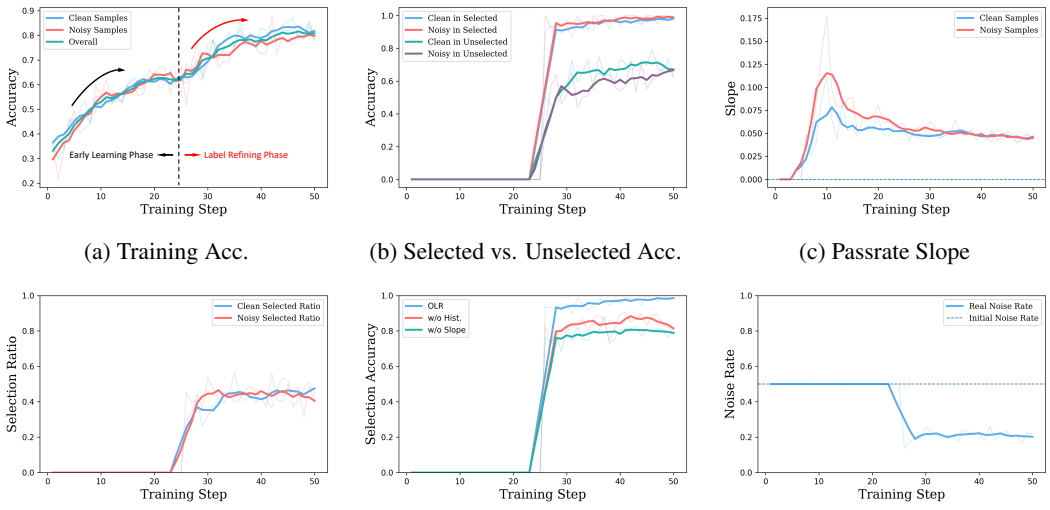


Figure 8: Results obtained with 50% inactive or active noisy labels in the High-Utilization Scenario using the Qwen-3-4B-Base model (Yang et al., 2025a). **For each sample, we take the majority vote across multiple model rollouts as the model's prediction.** The left and right pairs of columns illustrate the model's predictions on clean and noisy samples, respectively, under both inactive and active noise. For clean samples, we display the proportion of correct (green) and incorrect (blue) predictions. For noisy samples, we show the proportion of predictions that are correct (green), that match the noisy label (red), or that match neither the true nor the noisy label (blue). The early training phase reveals that the model learns to predict true labels even on noisy examples, indicating a preference for fitting correctly labeled samples and an increasing likelihood of producing correct answers on noisy ones over time.

Table 7: Results on Qwen3-8B-Base under the High-Utilization Scenario and 50% noise ratio. **Bold** indicates the better.

Method	In-Distribution							Out-of-Distribution			
	AIME24	AIME25	AMC	MATH-500	Minerva	Olympiad	**Avg.**	ARC-c	GPQA*	MMLU-Pro	**Avg.**
Qwen3-8B-Base	10.0	7.5	38.0	61.2	27.2	33.3	29.5	28.8	13.6	46.9	29.8
Active Noisy Label											
w/o OLR	14.6	12.5	52.3	80.2	39.7	43.9	40.5	59.6	19.2	60.3	46.4
w/ OLR	16.3	15.0	50.3	80.4	42.6	45.5	**41.7**	75.5	25.3	60.1	**53.6**
Inactive Noisy Label											
w/o OLR	24.6	20.4	58.3	85.0	39.7	52.7	46.8	67.2	34.3	62.9	54.8
w/ OLR	24.6	18.8	56.8	86.6	44.1	51.6	**47.1**	75.5	31.3	62.2	**56.3**

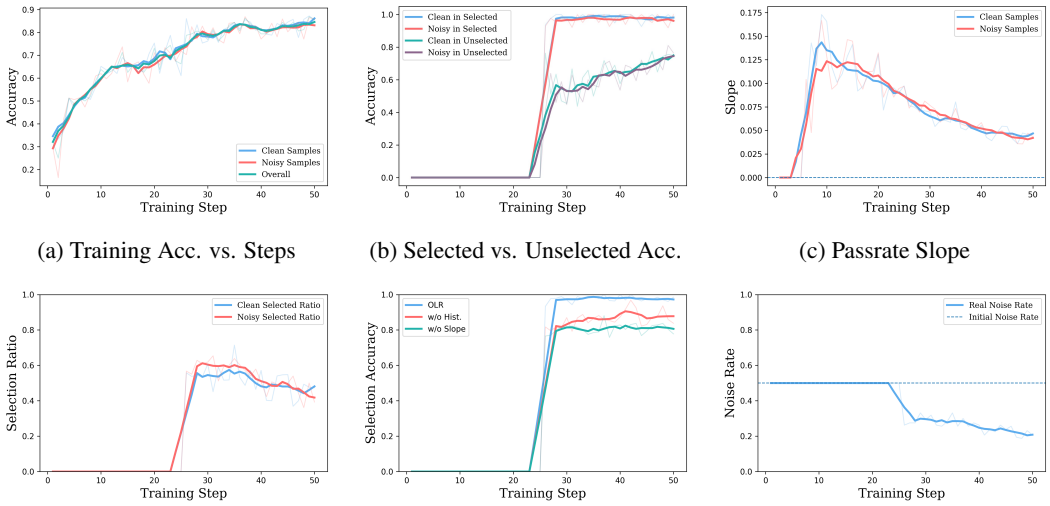
pass rates indicate greater difficulty. As shown in Figure 7, the DAPO-Math dataset contains a total of 14,116 samples. The difficulty distribution exhibits a long-tail pattern as the pass rate increases (i.e., as the difficulty decreases). Among these, approximately 40% (5,688/14,116) are too difficult (pass rate: 0.0). These extremely challenging samples are detrimental to RLVR training, as the model receives too few effective supervision signals, resulting in slow model updates. Conversely, if the samples are too easy, the model is unlikely to sample enough erroneous reasoning paths, hindering effective model training and updates. Recent work (Qu et al., 2026; Zhan et al., 2025) indicates that questions of moderate difficulty are most conducive to effective RLVR training. To minimize confounding factors and accurately assess the impact of noisy labels, we follow this principle in our data selection. Specifically,



(a) Training Acc. (b) Selected vs. Unselected Acc. (c) Passrate Slope

(d) Selection Ratio (e) Ablation Acc. (f) Real vs. Initial Noise

Figure 9: Training dynamics of OLR with Qwen3-4B-Base under the active noise setting ($\rho = 0.5$) and the High-Utilization Scenario. Acc. denotes accuracy.



(a) Training Acc. vs. Steps (b) Selected vs. Unselected Acc. (c) Passrate Slope

(d) Selection Ratio (e) Ablation Acc. (f) Real vs. Initial Noise

Figure 10: Training dynamic of OLR with Qwen3-4B-Base under the inactive noisy scenario ($\rho = 0.5$).

Table 8: Results on Deepseek-R1-Distill-Llama-8B under the High-Utilization Scenario and 50% noise ratio. **Bold** indicates the better.

Method	In-Distribution							Out-of-Distribution			
	AIME24	AIME25	AMC	MATH-500	Minerva	Olympiad	**Avg.**	ARC-c	GPQA*	MMLU-Pro	**Avg.**
Deepseek-R1-Distill-Llama-8B	15.8	17.9	48.2	72.2	18.8	35.7	34.8	24.9	10.1	38.7	24.6
Active Noisy Label											
w/o OLR	21.7	19.6	56.2	72.2	18.4	40.4	38.1	20.6	13.1	41.7	25.1
w/ OLR	25.8	17.5	60.5	73.8	18.4	39.3	**39.2**	27.2	14.6	41.5	**27.8**
Inactive Noisy Label											
w/o OLR	28.9	22.9	65.8	80.0	22.8	45.2	44.2	31.0	18.7	45.0	31.6
w/ OLR	34.2	20.4	65.1	80.0	24.3	45.6	**44.9**	33.8	19.2	45.0	**32.7**

∘ For `High-Utilization Scenario`, we construct a dataset by collecting samples with a pass rate of 0.5 from the DAPO-Math dataset using the Qwen3-4B-Base model. This setting is designed to eliminate the impact of problem difficulty and isolate the most critical samples with maximum

informational value for RLVR training, where every sample plays a pivotal role in training dynamics. By doing so, we aim to investigate the effect of noisy labels on RLVR training in the cleanest possible environment. As shown in Figure 7, there are a total of 866 samples with a pass rate of 0.5. To facilitate integer divisibility for mini-batch processing (e.g., batch size = 160), we randomly selected 800 of them to construct the training dataset. In this scenario, we conduct a comprehensive analysis under different noise ratios (0.1, 0.3, 0.5, 0.7, 0.9) and compare the results with various baseline methods.

◦ For `Realistic Mixed-Difficulty Scenario`, we construct a dataset by collecting samples with pass rates between 0.25 and 0.75 from the DAPO-Math dataset using the Qwen3-4B-Base model. This setup is intended to better reflect the broad range of training sample difficulties found in real-world scenarios, allowing us to investigate the impact of noisy labels under realistic conditions. As shown in Figure 7, there are a total of 4,794 samples with a pass rate between 0.25 and 0.75. To facilitate integer divisibility for mini-batch processing (e.g., batch size = 160), we randomly selected 4,000 of them to construct the training dataset. It is worth noting that the Qwen3 technical report (Yang et al., 2025a) shows that the Qwen3 model uses only 3,995 samples for RLVR training, indicating that RLVR does not necessarily require a large number of samples. Our choice of 4,000 samples is already sufficient to cover common training scenarios. We find that the performance gain of the model trained on these 4,000 samples under clean conditions is very limited compared to that of the model trained on 800 moderately difficult samples that are most beneficial for training. This further demonstrates that ***in RLVR training, sample quality is more important than sample quantity***. In practice, since samples are often carefully selected, the proportion of noisy labels is typically not very high, so we analyze the effects at noise ratios of 0.2 and 0.5.

For inactive noise injection, we replace correct labels with a string that the model is unlikely to output, specifically "`__NoCorrectAnswer__`". Empirical verification shows that the proportion of boxed answers output by the model on the DAPO-Math dataset that match this string is consistently zero.

Fixed wrong labels can be used for an active-noise control, but a nominal corruption ratio does not maintain the same effective active ratio: appearance of each wrong label in a rollout group depends on the current policy and K. For example, if 500 of 1000 prompts have fixed wrong labels with model probability 0.5, the probability that all 500 labels appear at least once in eight rollouts is only $(1 - 0.5^8)^{500} \approx 0.14$. The observed probabilities of wrong labels are often lower (about 20–30%; Figure 8). Thus, fixed labels provide a useful robustness check (Table 13), while dynamic construction controls the effective active-noise rate in our main experiments.

Thus, we dynamically construct active noise labels during inference by replacing correct labels with incorrect answers generated by the model in an on-policy manner. Specifically, for samples labeled as strongly noisy, during training, we randomly select an incorrect answer from multiple model rollouts to replace the "`__NoCorrectAnswer__`" string. Notably, this implies that for training sets with the same proportion of inactive and active noise, the set of samples we mark as noisy is identical; the only difference is that inactive noise labels remain fixed as the "`__NoCorrectAnswer__`" string, whereas active noise labels are replaced during training with incorrect answers generated by the model itself, thereby constructing a more challenging noisy-label training set.

C.2 Implementation Details

We use Qwen3-4B-Base (Yang et al., 2025a), Qwen3-8B-Base (Yang et al., 2025a), and Deepseek-R1-Distill-Llama-8B (Grattafiori et al., 2024) as the training models. All experiments were conducted on 8 × NVIDIA A100 (80G) GPUs. The validation and test sets followed those used in prior work (Yan et al., 2025). We set the hyperparameters to data.train_batch_size $= 160$, trainer.total_epochs $= 10$, with a maximum response length of 4096 and 8 rollouts. During training, we used a learning rate of 1e-6 and a KL loss weight of 1e-3. The temperature coefficient was set to 1.0 during training, and to 0.6 for both validation and testing. We use the validation set (Yan et al., 2025) to tune the parameters δ_{slope} and T, and to select the best checkpoint for each method, comparing their performance. Our implementation is built upon veRL[1], which uses vLLM[2] as the rollout generator. We thank the authors for making these open-source repositories available.

[1] https://github.com/volcengine/verl
[2] https://github.com/vllm-project/vllm

Each full RLVR training run requires on-policy generation of eight responses per prompt, with responses up to 4096 tokens, over ten epochs on eight A100 GPUs. Repeating the full sweep across training seeds would multiply this cost, so we prioritized matched GRPO–OLR comparisons across noise ratios, noise mechanisms, and model backbones, together with the fixed natural-noise evaluation. These comparisons probe consistency across settings rather than seed-to-seed variation: reported scores are individual training outcomes, and small numerical differences should be interpreted descriptively rather than as statistically resolved gains.

C.3 Baseline Description

We compare two categories of baseline methods. The first category consists of unsupervised approaches that do not rely on ground-truth labels:

- **TTRL** (Zuo et al., 2025): uses majority-voted outputs as pseudo-labels;
- **Co-Reward** (Zhang et al., 2025b): leverages contrastive consistency to alleviate collapse in unsupervised RLVR training;
- **Self-Certainty** (Zhao et al., 2025): promotes high-confidence predictions by maximizing KL divergence;
- **Token-Level Entropy** (Agarwal et al., 2025): enforces consistency by minimizing token-level entropy;
- **Sentence-Level Entropy** (Agarwal et al., 2025): encourages coherent predictions by maximizing sentence-level likelihood.

The second category includes noise-robust learning methods and transfer-friendly regularization techniques:

- **Confidence Penalty** (Pereyra et al., 2017): discourages overconfident predictions to reduce overfitting to noisy labels (implemented at the token level in our setting);
- **Label Smoothing** (Lukasik et al., 2020): replaces one-hot targets with smoothed distributions to mitigate overconfidence and noise sensitivity (also applied at the token level);
- **Small-loss Selection** (Gui et al., 2021): treats low-loss samples as clean and prioritizes them for training;
- **Random Selection**: randomly samples training instances as a control variant.

For all baselines, we use the same parameters as in OLR. For method-specific parameters, we follow the values given in their respective papers. We then use the same validation set to select the best checkpoint for each method and compare their performance.

D More Experiments

D.1 OLR Performance on Models of Different Sizes and Types

We run OLR on Qwen3-8B-Base. The experiments in Table 7 show that OLR also delivers performance gains, particularly on OOD benchmarks, where it achieves an average improvement of **4.4%** over models trained without OLR across both noise scenarios. At the same time, the results shown in Table 8 demonstrate that OLR improves performance on Deepseek-R1-Distill-Llama-8B (Grattafiori et al., 2024) across both ID and OOD benchmarks. This clearly demonstrates the versatility of OLR.

D.2 Additional Controls for Label Refinement

We report matched controls on Qwen3-4B-Base in the High-Utilization setting. Unless stated otherwise, the injected noise ratio is $\rho = 0.5$, and ID and OOD values are average benchmark accuracies in percent.

Table 9: Results on Qwen3-4B-Base under the High-Utilization Scenario and 50% noise ratios with different slope threshold δ_{slope}. **Bold** indicates the best.

Method	In-Distribution							Out-of-Distribution			
	AIME24	AIME25	AMC	MATH-500	Minerva	Olympiad	**Avg.**	ARC-c	GPQA*	MMLU-Pro	**Avg.**
Qwen3-4B-Base	9.6	4.2	34.2	52.6	19.5	28.4	24.7	35.8	14.1	33.3	27.7
Active Noisy Label											
GRPO	10.4	8.8	45.8	74.6	33.1	40.4	35.5	73.9	24.7	48.6	49.1
OLR w/ $\delta_{\text{slope}} = 0.0$	13.3	9.2	46.8	76.4	38.2	40.7	37.4	85.4	30.8	56.8	**57.7**
OLR w/ $\delta_{\text{slope}} = 0.025$	10.0	10.0	43.1	76.0	34.2	38.8	35.4	73.5	27.3	50.2	50.3
OLR w/ $\delta_{\text{slope}} = 0.050$	20.4	15.4	49.7	81.4	36.4	48.1	**41.9**	78.5	28.3	54.8	53.9
OLR w/ $\delta_{\text{slope}} = 0.075$	12.5	11.3	45.8	75.4	36.2	39.4	36.8	82.6	31.3	56.4	56.8
OLR w/ $\delta_{\text{slope}} = 0.1$	10.8	8.8	44.3	76.4	37.5	41.8	36.6	80.0	26.8	54.1	53.6
OLR w/ $\delta_{\text{slope}} = 0.2$	9.6	8.8	46.1	76.2	40.1	38.8	36.6	73.0	26.8	54.2	51.3
Inactive Noisy Label											
GRPO	12.5	5.0	45.3	76.0	34.9	42.7	36.1	86.7	28.8	56.0	57.2
OLR w/ $\delta_{\text{slope}} = 0.0$	12.9	9.2	48.0	78.2	40.1	41.8	38.4	88.0	28.3	56.1	57.5
OLR w/ $\delta_{\text{slope}} = 0.025$	12.9	15.0	47.1	79.8	43.4	43.1	40.2	85.6	30.8	56.6	57.7
OLR w/ $\delta_{\text{slope}} = 0.050$	21.7	18.3	53.9	84.2	42.3	48.9	**44.9**	86.8	38.4	57.8	**61.0**
OLR w/ $\delta_{\text{slope}} = 0.075$	14.2	9.6	48.5	79.2	40.4	44.4	39.4	82.6	31.3	56.4	56.8
OLR w/ $\delta_{\text{slope}} = 0.1$	10.8	12.1	48.3	78.0	39.7	41.5	38.4	85.9	29.3	56.2	57.1
OLR w/ $\delta_{\text{slope}} = 0.2$	15.8	5.4	47.1	77.8	39.3	40.7	37.7	85.8	27.8	54.9	56.2

Table 10: Results on Qwen3-4B-Base under the High-Utilization Scenario and 50% noise ratios with different lengths of the early learning stage T. **Bold** indicates the best.

Method	In-Distribution							Out-of-Distribution			
	AIME24	AIME25	AMC	MATH-500	Minerva	Olympiad	**Avg.**	ARC-c	GPQA*	MMLU-Pro	**Avg.**
Qwen3-4B-Base	9.6	4.2	34.2	52.6	19.5	28.4	24.7	35.8	14.1	33.3	27.7
Active Noisy Label											
GRPO	10.4	8.8	45.8	74.6	33.1	40.4	35.5	73.9	24.7	48.6	49.1
OLR w/ $T = 2$	11.3	9.6	46.0	75.8	37.1	42.5	37.1	83.4	28.3	55.4	55.7
OLR w/ $T = 4$	20.4	15.4	49.7	81.4	36.4	48.1	**41.9**	78.5	28.3	54.8	53.9
OLR w/ $T = 6$	9.6	10.0	45.9	77.2	39.7	40.0	37.1	79.7	26.8	53.8	53.4
OLR w/ $T = 8$	12.1	7.1	43.7	76.2	37.5	41.2	36.3	84.7	31.3	53.2	**56.4**
Inactive Noisy Label											
GRPO	12.5	5.0	45.3	76.0	34.9	42.7	36.1	86.7	28.8	56.0	57.2
OLR w/ $T = 2$	11.7	13.3	47.1	79.4	40.1	42.1	39.0	87.7	31.3	57.3	58.8
OLR w/ $T = 4$	21.7	18.3	53.9	84.2	42.3	48.9	**44.9**	86.8	38.4	57.8	**61.0**
OLR w/ $T = 6$	19.2	17.9	52.9	85.0	40.1	49.0	44.0	82.3	33.8	57.9	58.0
OLR w/ $T = 8$	12.1	7.5	47.1	79.6	41.5	42.5	38.4	85.2	33.8	55.9	58.3

Table 11: Ablation results on Qwen3-4B-Base under the High-Utilization Scenario and 50% noise ratio. **Bold** indicates the best.

Method	In-Distribution							Out-of-Distribution			
	AIME24	AIME25	AMC	MATH-500	Minerva	Olympiad	**Avg.**	ARC-c	GPQA*	MMLU-Pro	**Avg.**
Active Noisy Label											
OLR	20.4	15.4	49.7	81.4	36.4	48.1	**41.9**	78.5	28.3	54.8	**53.9**
- Positive Convergence Slope	10.8	7.9	46.7	79.0	35.7	42.1	37.0	73.9	24.7	55.5	51.4
- Historical Consistency	13.8	10.4	46.2	77.2	35.3	40.4	37.2	78.5	26.8	54.5	53.3
Inactive Noisy Label											
OLR	21.7	18.3	53.9	84.2	42.3	48.9	**44.9**	86.8	38.4	57.8	**61.0**
- Positive Convergence Slope	10.0	10.8	47.4	79.6	39.7	42.1	38.3	86.4	25.3	56.2	56.0
- Historical Consistency	10.0	6.7	47.1	78.0	38.6	41.0	36.9	85.1	31.3	54.0	56.8

Unconditional majority relabeling and zero-variance filtering. Always-relabel uses the same warm-up as OLR and then replaces each reference label with the current rollout majority without either OLR gate. Table 12 shows that it yields only small ID gains over noisy GRPO and lowers inactive OOD accuracy from 57.2% to 54.7%. Full OLR improves active and inactive ID accuracy by 6.4 and 8.8 points over noisy GRPO. The single-gate ablations in Table 11 further show that both gates contribute to this difference. Filtering zero-variance rollout groups changes GRPO's ID accuracy by at most 0.3 points and does not close the gap to OLR. Thus, neither delayed majority replacement nor zero-variance filtering explains the observed gains.

Fixed active labels and clean supervision. For fixed active noise, we generate incorrect boxed answers with the base model once, replace the correct labels on 50% of prompts, and keep those wrong labels unchanged throughout training. This differs from the dynamic active-noise setting, which selects a wrong answer from the current rollout group each time a prompt is revisited. Table 13 shows that OLR gains 5.2 ID and 3.5 OOD points over GRPO with fixed active labels. Fixed corruption is milder than dynamic corruption: although 50% of prompts receive wrong labels, only about 20% of the pool is effectively active during training, and the injected wrong answer matches

Table 12: Additional baselines under active and inactive noise ($\rho = 0.5$). All values are average accuracy (%).

Method	Active ID	Inactive ID	Active OOD	Inactive OOD
GRPO	35.5	36.1	49.1	57.2
Always-relabel	36.3	36.6	50.2	54.7
GRPO + zero-variance filtering	35.3	36.4	48.9	56.4
OLR	**41.9**	**44.9**	**53.9**	**61.0**

Table 13: Fixed versus dynamic active noise ($\rho = 0.5$) and a clean-label control ($\rho = 0$). All values are average accuracy (%).

Labels	Method	ID	OOD
Fixed active	GRPO	37.1	54.7
Fixed active	OLR	**42.3**	**58.2**
Dynamic active	GRPO	35.5	49.1
Dynamic active	OLR	**41.9**	**53.9**
Clean	GRPO	44.5	60.8
Clean	OLR	**44.9**	**61.5**

Table 14: Rollout feasibility of 107 naturally noisy BIRD training labels, using five seeds and eight rollouts per prompt.

Model	Active	Inactive	Est. active count
Qwen3-4B-Base	18%	82%	19
Qwen3-8B-Base	14%	86%	15
Olmo-3-1025-7B	9%	91%	10

the rollout majority on roughly 20–30% of corrupted prompts under $K = 8$. With no injected noise, OLR scores 44.9% ID and 61.5% OOD versus 44.5% and 60.8% for GRPO, indicating no aggregate degradation from refinement on clean supervision.

D.3 Natural-Noise Analysis on BIRD Text2SQL

We construct the training set from paired BIRD-600-Original and BIRD-600-Corrected examples, retaining a pair only when its question and evidence are unchanged and both SQL queries are executable. This yields 348 training examples. Of these, 107 original SQL labels produce execution results different from their corrected counterparts (30.7%); the remaining 241 examples are also retained for training. The evaluation set comprises 189 disjoint, human-corrected, executable examples spanning 58 databases. To estimate whether these wrong labels are reachable by the base policy, we use five rollout seeds and eight rollouts per prompt. We classify a wrong label as active if any rollout's execution result matches the original SQL. Table 14 shows that 9–18% of these natural errors are active under this finite-rollout criterion; most are inactive, but the data contain both failure modes.

Table 15 gives database-level examples from the 189-example BIRD test set. The examples show improvements on distinct databases for all three backbones; the overall scores in Table 5 remain the appropriate summary across all 58 databases. Since each database contains few test examples, the large percentage changes in individual rows should be read alongside those overall scores.

D.4 Detailed Analysis on Early Correctness Coherence

To verify the Early Correctness Coherence phenomenon, we conduct a detailed analysis of the model's predictions on clean and noisy samples under both noise scenarios (see Figure 8). The results substantiate the two key assumptions of Theorem 2.3: (i) correct labeling probability is higher for clean samples than for noisy ones, reflecting the quality of pre-training; and (ii) even under noisy supervision, the probability of correct answers for noisy samples increases during early training. These empirical observations lay a solid foundation for the proposed OLR method.

Table 15: Examples of BIRD database-level execution accuracy (%). The table lists selected databases, not all 58 test databases.

Model	Database	Noisy GRPO	OLR
Qwen3-4B-Base	address	42.86	**57.14**
	beer_factory	25.00	**50.00**
	cars	80.00	**100.00**
	legislator	25.00	**50.00**
	movie_3	40.00	**60.00**
	simpson_episodes	25.00	**50.00**
	world	0.00	**66.67**
Qwen3-8B-Base	cs_semester	33.33	**100.00**
	law_episode	50.00	**100.00**
	menu	33.33	**66.67**
	movie_platform	40.00	**60.00**
Olmo-3-1025-7B	computer_student	33.33	**66.67**
	law_episode	50.00	**100.00**
	chicago_crime	50.00	**75.00**
	books	60.00	**80.00**
	movie_platform	40.00	**60.00**
	olympics	0.00	**50.00**
	car_retails	0.00	**20.00**

D.5 Training Dynamics with Inactive Noisy Labels

Figure 10 shows training dynamics under 50% inactive noise. In (a), early learning lifts majority accuracy for both clean/noisy samples from 50% to over 70%. Accuracy then plateaus until OLR boosts it further to over 80%, a **10%** gain showing OLR builds on early learning. In (b), OLR-selected samples exceed **90%** accuracy (approaching 100%), while unselected ones stay below 80%. (c) shows positive pass rate slopes throughout training for both sample types, indicating correct answers emerge with increasing probability on noisy samples via cross-sample coupling. (d) shows OLR selects over 40% of both clean/noisy sets, avoiding bias toward simple samples. (e) shows both criteria together achieve near 100% selection accuracy; removing either reduces accuracy by **20%**, highlighting their indispensability. Finally, (f) shows OLR reduces inactive noise by **25%** during refinement, demonstrating robustness.

E More Related Work

Noisy Label Learning is a mature field, but its techniques rarely accommodate RLVR's open-ended generation space and on-policy dynamics. In traditional classification scenarios, methods for mitigating noisy labels are abundant and typically fall into the following categories:

Noise Transition Matrix Estimation (Xiao et al., 2015; Chen & Gupta, 2015; Srivastava et al., 2014; Sukhbaatar et al., 2014; Patrini et al., 2017; Goldberger & Ben-Reuven, 2017; Lee et al., 2019; Yao et al., 2018; Han et al., 2018a; Hendrycks et al., 2018; Collier et al., 2021; Bucarelli et al., 2023): These approaches generally involve learning the flipping probabilities between true and noisy labels, mimicking the noise process. However, this is unsuitable for RLVR scenarios because noisy labels in such contexts often form an open set without inherent class boundaries, making it difficult to derive a noise transition matrix.

Loss Correction (Goodfellow et al., 2014; Pereyra et al., 2017; Zhang et al., 2017; Menon et al., 2020; Hendrycks et al., 2019; Tanno et al., 2019; Jenni & Favaro, 2018; Zhang et al., 2021a; Wei et al., 2021; Lukasik et al., 2020; Xia et al., 2020; Qu et al., 2021; Cheng et al., 2021): These methods typically employ regularized algorithms to reduce model overfitting to noisy labels. While primarily designed for classification tasks, RLVR involves generative tasks, which limits the direct applicability of most existing methods.

Small-loss-based Sample Selection (Malach & Shalev-Shwartz, 2017; Wang et al., 2018; Han et al., 2018b; Jiang et al., 2018; Huang et al., 2019; Chen et al., 2019; Shen & Sanghavi, 2019; Yu et al., 2019; Wu et al., 2020; Wei et al., 2020; Wu et al., 2021; Song et al., 2021): These approaches identify

samples with low training loss as potentially clean, exploiting the phenomenon of early memorization. However, in RLVR, small-loss samples are not necessarily beneficial. For instance, when a model's rollout is either entirely correct or entirely incorrect for a sample, its loss is zero. Yet such samples offer no training value and should be avoided (Zhan et al., 2025).

Semi-supervised Learning-based Methods (Berthelot et al., 2019; Song et al., 2019; Nguyen et al., 2019; Zhou et al., 2020; Li et al., 2020; Kim et al., 2021b; Zhang et al., 2021b; Bai et al., 2021; Gui et al., 2021; Kim et al., 2021a; Ortego et al., 2021; Yao et al., 2021; Wang et al., 2022; Karim et al., 2022; Li et al., 2022; Chen et al., 2023; Li et al., 2023; Feng et al., 2023; Huang et al., 2023): While abundant in traditional classification, semi-supervised methods are task-specific and scarce in RLVR, where pre-selecting clean samples from noisy data remains an unsolved challenge that prevents direct transfer.