TTCL: TEST-TIME CALIBRATION LEARNING FOR LARGE LANGUAGE MODEL REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Reliable large language models (LLMs) must not only produce accurate answers but also express confidence that faithfully reflects their probability of being correct. Such calibration is essential for identifying uncertain predictions and supporting reliable decision-making in real-world deployment. Recent studies incorporate calibration learning into reinforcement learning (RL), jointly optimizing answer correctness and verbalized confidence using ground-truth correctness supervision. However, their reliance on labeled data limits their applicability in practical test-time settings, where ground-truth labels are unavailable and calibration may need to adapt to newly encountered target tasks. To address this challenge, we propose *Test-Time Calibration Learning (TTCL)*, a label-free framework that jointly adapts reasoning accuracy and verbalized confidence directly on unlabeled target-task data. Specifically, TTCL derives self-supervision signals for both correctness and calibration from multiple model-generated responses, enabling calibration learning at test time without ground-truth labels. Theoretical analysis further establishes TTCL as a bounded surrogate for the ideal calibration objective. Extensive experiments on mathematical reasoning and factual question answering demonstrate that TTCL consistently improves both accuracy and calibration across diverse models and tasks. On base models, TTCL achieves an average relative accuracy improvement of $+40.13\%$ and an ECE reduction of $+70.80\%$ across eight benchmarks. Moreover, TTCL can further improve both accuracy and calibration for already calibrated models under domain shift, particularly when source-domain calibration transfers poorly to target tasks. In the math-to-factQA setting, TTCL achieves an average relative accuracy gain of $+20.35\%$ and reduces ECE by $+53.83\%$.

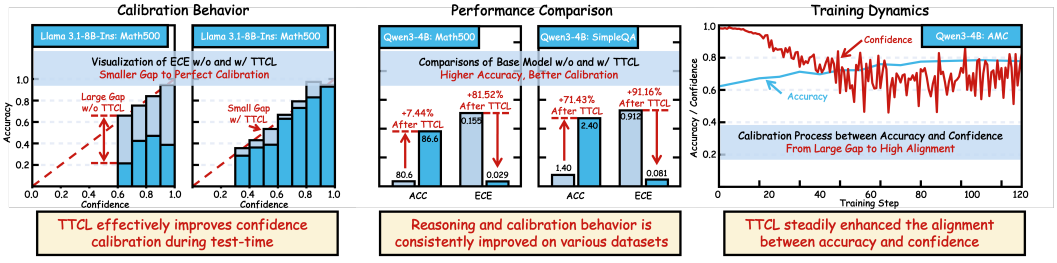


Figure 1: **Overview of our work**: TTCL enables label-free calibration learning directly on unlabeled target-task data, consistently improving reasoning accuracy while reducing calibration error across models and tasks, with confidence progressively aligning with accuracy during test-time learning.

1 INTRODUCTION

Large language models (LLMs) are increasingly deployed in user-facing and decision-support systems (Zhang et al., 2026; Sun et al., 2026), where the outputs may directly influence real-world decisions. Despite their remarkable capabilities, LLMs often assign high confidence to incorrect responses. Such miscalibration makes errors difficult to detect and can lead to severe consequences in high-stakes settings (Ji et al., 2023). Ideally, a well-calibrated model should produce confidence estimates that faithfully reflect its empirical likelihood of being correct. To achieve this, researchers have proposed calibration learning (Damani et al., 2026; Bani-Harouni et al., 2026), which combine

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.

calibration-oriented reward with answer correctness reward during reinforcement learning process to consistently improve LLM reasoning performance and confidence calibration behavior.

Despite the promising results achieved by previous research, we observe that existing calibration learning methods typically rely on large amounts of labeled data to provide supervision for calibration, incurring substantial annotation costs (Damani et al., 2026; Tan et al., 2026; Zollo et al., 2026). More importantly, a model calibrated on a source domain may not remain well-calibrated when testing on a different target domain. As shown in Fig. 2, a model calibrated by existing method RLCR (Damani et al., 2026) on DAPO-Math-14K (Yu et al., 2025) transfers well to the math-related AMC benchmark (Yang et al., 2024), but remains limited calibration behavior on the factual NQ benchmark (Kwiatkowski et al., 2019), highlighting the sensitivity of learned calibration to domain shift. These limitations pose a key challenge: *how can we adapt a model's calibration behavior directly to an unseen target task at test time, without relying on ground-truth labels?*

To address this, we propose ***T**est-**T**ime **C**alibration **L**earning* (TTCL), which provides a label-free surrogate for calibration learning. Specifically, TTCL derives two self-supervision signals from multiple sampled responses: a correctness reward based on the majority answer and a calibration reward based on the empirical support of each answer. The two rewards are separately applied to the answer and confidence tokens, allowing the model to improve answer correctness and confidence calibration without undesirable interference. To stabilize learning, we further introduce a target cache that maintains temporally lagged supervision targets, reducing the risk of reinforcing the model's own instantaneous predictions. To demonstrate the effectiveness of our method, we conduct extensive experiments on mathematical reasoning and factual question-answering benchmarks with multiple model scales and series. The results show that TTCL can consistently improve both reasoning accuracy and confidence on target task. Our main contributions are summarized as follows:

- Conceptually, we propose Test-Time Calibration Learning, a new calibration paradigm that adapts LLM confidence directly to the target task at test time without relying on ground-truth labels, enabling calibration of confidence estimation under unseen test distributions (Section 3.1).
- Technically, we develop a label-free TTCL framework that constructs self-supervised signals for correctness and calibration rewards from multi-rollout frequency (Section 3.2 & 3.3).
- Theoretically, we show that TTCL essentially serves as a bounded surrogate for the ideal calibration learning objective, theoretically grounding its label-free learning mechanism (Section 4).
- Experimentally, extensive experiments across models and benchmarks show consistent improvements of TTCL in both reasoning and calibration performance (Section 5).

2 PRELIMINARY

In this section, we first introduce the problem setup of calibration learning and the corresponding calibration measurement, and then review Group Relative Policy Optimization (GRPO), a widely used RL algorithm in eliciting LLM reasoning capability.

Problem setup of calibration learning. Let π_θ denote an LLM parameterized by θ. Given a question x, the model generates a response $y \sim \pi_\theta(\cdot|x)$ containing a reasoning trajectory and an extracted final answer $a = \mathrm{Ans}(y)$. Given the ground-truth (GT) answer a^*, the correctness indicator is defined as $z(y, a^*) = \mathbb{I}[a \equiv a^\star] \in \{0, 1\}$. A confidence estimator $c(x, y) \in [0, 1]$ quantifies the model's belief that the generated response y is correct. Let C and Z denote the corresponding confidence and correctness random variables. A perfectly calibrated model satisfies:

$$\Pr(Z = 1|C = c) = c, \tag{1}$$

indicating that predictions with confidence c should be correct about c fraction of the time. In practice, calibration learning typically optimizes $c(x, y)$ against observed correctness $z(y)$ using a proper scoring rule (Gneiting & Raftery, 2007). A common choice is the Brier score (Brier, 1950), $\mathcal{L}_{\mathrm{Brier}} = (c(x, y) - z(y, a^*))^2$, which encourages the predicted confidence to match the probability of correctness. Building on this, recent methods (Damani et al., 2026; Bani-Harouni et al., 2026; Ma et al., 2026) jointly optimize reasoning correctness and confidence calibration during RL. Specifically, RLCR (Damani et al., 2026) trains the LLM to verbalize a confidence score using reward as:

$$r_{\mathrm{RLCR}} = z(y, a^*) - (c(x, y) - z(y, a^*))^2, \tag{2}$$

where the first term is a binary correctness reward, while the second is a Brier-score calibration reward. However, both objectives require the ground-truth answer a^*, which limits their applicability

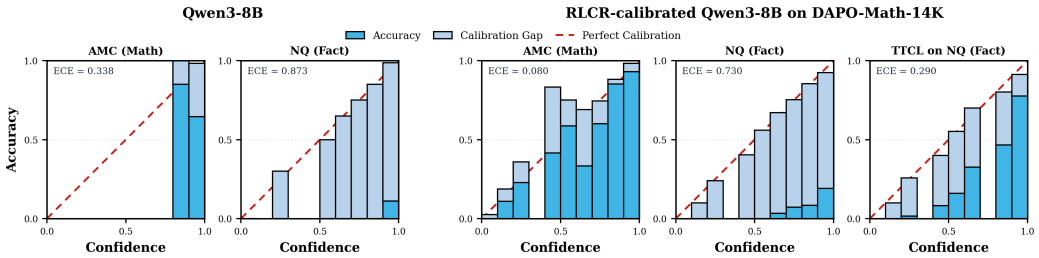


Figure 2: **Calibration performance of Qwen3-8B and its RLCR-calibrated model on DAPO-Math-14K.** *Left:* Base Qwen3-8B exhibits over-confidence on both AMC and NQ, with predicted confidence exceeding empirical accuracy. *Right:* RLCR on DAPO-Math-14K transfers well to AMC but remains limited on NQ, where further TTCL adaptation markedly improves calibration.

when annotated labels are not available. In this work, we focus on test-time calibration learning, where the model adapts its confidence estimation on unlabeled test data.

Calibration measurement. Expected Calibration Error (ECE) is a widely used metric for evaluating calibration by measuring the discrepancy between predicted confidence and empirical accuracy (Guo et al., 2017). Given N predictions partitioned into M confidence bins $\{B_m\}_{m=1}^{M}$, ECE is defined as:

$$\mathrm{ECE} = \sum_{m=1}^{M} \frac{|B_m|}{N} \left|\mathrm{Acc}(B_m) - \mathrm{Conf}(B_m)\right|, \tag{3}$$

where $\mathrm{Acc}(B_m)$ and $\mathrm{Conf}(B_m)$ denote the average correctness and confidence within bin B_m, respectively. A lower ECE indicates better calibration, with confidence better aligned with accuracy.

3 TTCL: Test-time Calibration Learning

In this section, we first present the motivation of the need for TTCL (Sec. 3.1), and then introduce the technical details of our proposed TTCL method (Sec. 3.2 and Sec. 3.3).

3.1 Why Test-time Calibration Learning (TTCL)?

Despite their strong reasoning capabilities, current LLMs often remain over-confident on downstream tasks. As illustrated in the left part of Fig. 2, Qwen3-8B exhibits clear over-confidence on both the mathematical benchmark AMC (Yang et al., 2024) and the factual benchmark NQ (Kwiatkowski et al., 2019), with estimated confidence systematically exceeding empirical accuracy and resulting in larger ECE values. Moreover, calibration learned on one task may not transfer well to others. As shown in the right part of Fig. 2, Qwen3-8B calibrated with RLCR on DAPO-Math-14K (Yu et al., 2025) achieves strong calibration on AMC, yet remains poorly calibrated on NQ. These observations motivate direct test-time calibration of LLMs and lead to two key principles of TTCL:

Key Principles of TTCL

- **Label-free calibration learning:** derive supervision signals for calibration learning from unlabeled test-time data, removing the dependence on annotated ground-truth labels.
- **Target-task adaptation:** adapt confidence estimation directly to newly encountered tasks at test time, alleviating the limited cross-task generalization of calibration learning.

Compared with existing calibration learning methods (Damani et al., 2026; Bani-Harouni et al., 2026), the first principle, *label-free calibration learning*, removes the dependence on annotated correctness labels, which are often costly to obtain or unavailable at test time. The second principle, *target-task adaptation*, enables the model to recalibrate its confidence directly on the deployment task, mitigating the limited cross-task transferability of calibration learned from source tasks. The right part of Fig. 2 further illustrates that TTCL can adapt calibration directly to a new target task at test time. Starting from the RLCR-calibrated Qwen3-8B, applying TTCL on unlabeled NQ data reduces the ECE from 0.730 to 0.290, without requiring additional ground-truth annotations. We next introduce how TTCL constructs label-free supervision for both correctness and calibration from unlabeled data.

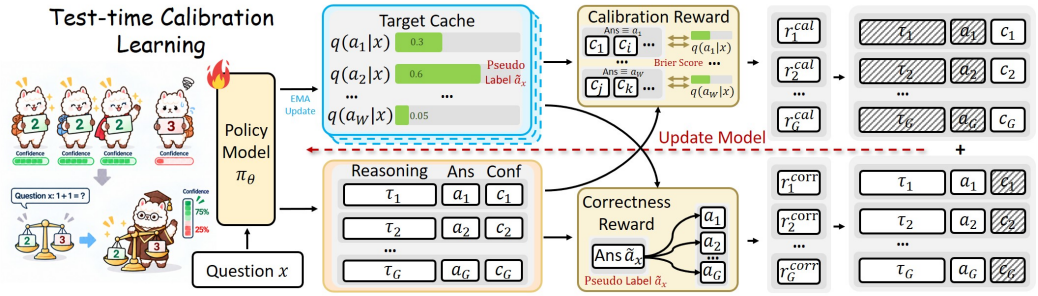


Figure 3: **Illustration of *TTCL* method**: Given an unlabeled question, TTCL samples multiple rollouts to construct label-free supervision: the most supported answer serves as the pseudo label, while the support of each answer provides its confidence target. A target cache stabilizes these signals and is updated via EMA across epochs. Correctness and calibration rewards are optimized separately, with the former updating reasoning and answer tokens and the latter updating only confidence tokens.

3.2 TTCL Self-Supervision Signals and Optimization Objective

Given a test-time question x, TTCL samples a group of G responses $\{y_i\}_{i=1}^{G}$. where each response is denoted as $y_i = (\tau_i, a_i, c_i)$, with τ_i denoting the reasoning trajectory, a_i the final answer, and $c_i \in [0, 1]$ the verbalized confidence. As illustrated in Fig. 3, our label-free supervision is built on a simple assumption: answers receiving stronger consensus across repeated sampled rollouts are more likely to be correct and should be associated with higher confidence. Based on this assumption, TTCL derives two self-supervision signals for correctness and calibration from the answer distribution.

Correctness self-supervision signal. Let $q(a|x)$ denote the empirical answer distribution induced by the rollout group. TTCL selects the most supported answer as the pseudo correctness target:

$$\tilde{a}_x = \arg\max_a q(a \mid x), \quad q(a \mid x) = \frac{1}{G}\sum_{j=1}^{G} \mathbb{I}[a_j \equiv a]. \tag{4}$$

The correctness reward for response y_i is then defined as:

$$r_i^{\text{corr}} = \mathbb{I}[a_i \equiv \tilde{a}_x] \in \{0, 1\}. \tag{5}$$

This majority-based pseudo-labeling strategy follows prior self-supervised RL methods (Zuo et al., 2025; Zhang et al., 2026; Shafayat et al., 2025), which use the most frequent answer as a pseudo correctness target to encourage consensus and improve LLM reasoning capability.

Calibration self-supervision signal. Unlike the correctness signal that only uses the most supported answer, calibration requires finer-grained supervision to reflect the model's uncertainty over different candidate answers. TTCL therefore uses the empirical support of each generated answer as its confidence target. For response y_i, the calibration target is defined as:

$$\tilde{p}_i = q(a_i \mid x). \tag{6}$$

Accordingly, responses whose answers receive stronger consensus across the rollout group are assigned higher confidence targets, while less-supported answers receive lower confidence targets. Given the predicted confidence c_i, the calibration reward is defined using the negative Brier score:

$$r_i^{\text{cal}} = -(c_i - \tilde{p}_i)^2 \tag{7}$$

In this way, TTCL converts the empirical answer frequency into a soft probability target for confidence calibration, enabling calibration learning without access to ground-truth correctness labels.

TTCL optimization objective. As the response y_i consists of the tokens of reasoning trajectory, final answer, and confidence prediction, naively combining the correctness and calibration rewards over the entire response can introduce undesirable interference between correctness and calibration (Ma et al., 2026). For example, correct reasoning and answer tokens may still be penalized due to poorly calibrated confidence. TTCL computes separate group-relative advantages for the two rewards:

$$\hat{A}_i^{\text{corr}} = r_i^{\text{corr}} - \frac{1}{G}\sum_{j=1}^{G} r_j^{\text{corr}}, \quad \hat{A}_i^{\text{cal}} = r_i^{\text{cal}} - \frac{1}{G}\sum_{j=1}^{G} r_j^{\text{cal}}. \tag{8}$$

The correctness advantage is applied only to the reasoning and answer tokens, while the calibration advantage is applied only to the confidence tokens. Accordingly, the TTCL objective is defined as:

$$\mathcal{J}_{\text{TTCL}}(\theta) = \mathbb{E}\left[\frac{1}{G}\sum_{i=1}^{G}\left[\frac{1}{|o_i^{\text{corr}}|}\sum_{y_t \in o_i^{\text{corr}}}\left(\rho_{i,t}(\theta)\hat{A}_i^{\text{corr}}\right) + \frac{1}{|o_i^{\text{cal}}|}\sum_{y_t \in o_i^{\text{cal}}}\left(\rho_{i,t}(\theta)\hat{A}_i^{\text{cal}}\right)\right]\right], \quad (9)$$

where $o_i^{\text{corr}} = (\tau_i, a_i)$ denotes the tokens of the reasoning trajectory and final answer, and $o_i^{\text{cal}} = (c_i)$ denotes the confidence tokens. The $\rho_{i,t}(\theta)$ denotes the PPO-style clipped importance ratio.

3.3 Target Cache for Stable Learning

The self-supervision signals above are constructed from the empirical answer distribution $q(a \mid x)$ of the current rollout group. However, directly using this instantaneous distribution can tightly couple the policy with its self-generated targets, leading to reward hacking. As majority-based supervision encourages answer consistency, their empirical frequencies rise, further increasing the associated confidence targets. Without stabilization, this feedback can drive the model toward a degenerate solution where rollouts collapse to the same incorrect answer with confidence approaching 1.0.

To mitigate this issue, TTCL introduces a *target cache* that maintains a temporally lagged answer distribution for each question. Let $q_{\text{cache}}^{(e)}(a \mid x)$ denote the cached answer distribution at epoch e. During epoch e, the correctness and calibration targets are constructed from the frozen cache rather than the current policy rollout distribution:

$$\tilde{a}_x^{(e)} = \arg\max_a q_{\text{cache}}^{(e)}(a \mid x), \quad \tilde{p}_i^{(e)} = q_{\text{cache}}^{(e)}(a_i \mid x). \quad (10)$$

The corresponding rewards are therefore:

$$r_i^{\text{corr}} = \mathbb{I}\left[a_i \equiv \tilde{a}_x^{(e)}\right], \quad r_i^{\text{cal}} = -\left(c_i - \tilde{p}_i^{(e)}\right)^2. \quad (11)$$

Before training, the cache is initialized from the answer distribution of the initial model. During training, the newly observed rollout distribution is not used for immediate supervision, but instead updates the cache for subsequent epochs. Let $q^{(e)}(a \mid x)$ denote the empirical answer distribution observed during epoch e, and the cache is updated using an exponential moving average (EMA):

$$q_{\text{cache}}^{(e+1)}(a \mid x) = (1-\eta)\, q_{\text{cache}}^{(e)}(a \mid x) + \eta\, q^{(e)}(a \mid x). \quad (12)$$

where η controls the update rate. By disentangling supervision targets from the current policy, the target cache blocks the positive feedback between answer consistency, confidence targets, and policy updates, stabilizing TTCL training and mitigating collapse toward a single high-confidence answer.

4 Theoretical Analysis of TTCL

In this section, we theoretically connect TTCL to the calibration definition in Eq. (1).

Analysis: *TTCL provides a label-free surrogate for calibration learning, with its deviation from the ideal calibration objective bounded by the frequency-based estimation error.*

Let $p^\star(x, y) = \Pr(Z = 1 \mid X = x, Y = y)$ denote the conditional probability that response y is correct for question x. For TTCL, we denote its frequency-based confidence target and the corresponding estimation error as $s(x, y) = q(\text{Ans}(y) \mid x)$ and $\epsilon = \mathbb{E}[|s(X, Y) - p^\star(X, Y)|]$ respectively.

Theorem 4.1 (TTCL as a Surrogate for the Ideal Calibration Objective). *Let $\mathcal{J}_{\text{TTCL}}(c) = \mathbb{E}\left[(c(X, Y) - s(X, Y))^2\right]$ denote the calibration objective optimized by TTCL, and let $\mathcal{J}_{\text{ideal}}(c) = \mathbb{E}\left[(c(X, Y) - p^\star(X, Y))^2\right]$ denote the ideal calibration objective. Then, for any confidence estimator $c(X, Y) \in [0, 1]$, $|\mathcal{J}_{\text{TTCL}}(c) - \mathcal{J}_{\text{ideal}}(c)| \le 2\epsilon$.*

Theorem 4.1 demonstrates that TTCL optimizes a bounded surrogate of the ideal calibration objective. The objective gap is controlled by the estimation error between the frequency-based target and the latent conditional correctness probability. A detailed proof is provided in Appendix D.3.

Table 1: **Performance of TTCL for base models on mathematical reasoning benchmarks.** Improved results after TTCL are highlighted in **bold**, with positive relative improvements shown in red color. AUROC is not reported for Llama-3.1-8B-Instruct on AIME25 because its accuracy is 0.

Methods	MATH500			AMC			AIME24			AIME25		
	ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑
Qwen3-4B												
Base Model	80.6	0.155	0.670	59.38	0.383	0.703	20.42	0.739	0.652	16.25	0.807	0.692
w/ **TTCL**	**86.6**	**0.029**	**0.866**	**77.81**	**0.091**	**0.910**	**28.33**	**0.065**	**0.856**	**20.42**	**0.261**	**0.878**
# Improv	+7.44%	+81.52%	+29.28%	+31.05%	+76.25%	+29.52%	+38.78%	+91.26%	+31.15%	+25.64%	+67.68%	+27.00%
Qwen3-8B												
Base Model	84.4	0.183	0.717	62.81	0.338	0.651	21.67	0.753	0.642	18.75	0.789	0.620
w/ **TTCL**	**89.0**	**0.131**	**0.830**	**80.63**	**0.168**	**0.844**	**30.83**	**0.120**	**0.882**	**26.67**	**0.145**	**0.908**
# Improv	+5.45%	+28.17%	+15.72%	+28.36%	+50.31%	+29.60%	+42.31%	+84.09%	+37.37%	+42.22%	+81.58%	+46.37%
Llama-3.1-8B-Instruct												
Base Model	39.6	0.436	0.479	21.88	0.611	0.518	4.58	0.807	0.365	0.00	0.853	-
w/ **TTCL**	**58.0**	**0.094**	**0.770**	**40.94**	**0.128**	**0.617**	**9.17**	**0.223**	**0.878**	0.00	**0.114**	-
# Improv	+46.46%	+78.50%	+60.99%	+87.14%	+79.12%	+19.03%	+100.00%	+72.37%	+140.72%	0.00%	+86.59%	-

Table 2: **Performance of TTCL for base models on factual QA benchmarks.** Improved results after TTCL are highlighted in **bold**, with positive relative improvements shown in red.

Methods	SimpleQA			NQ			HotpotQA			TriviaQA		
	ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑
Qwen3-4B												
Base Model	1.40	0.912	0.481	8.63	0.859	0.705	9.56	0.831	0.703	15.63	0.789	0.660
w/ **TTCL**	**2.40**	**0.081**	**0.731**	**10.63**	**0.155**	**0.848**	**10.06**	**0.367**	**0.816**	**17.00**	**0.257**	**0.766**
# Improv	+71.43%	+91.16%	+51.96%	+23.25%	+81.95%	+20.32%	+5.20%	+55.82%	+16.13%	+8.80%	+67.41%	+16.08%
Qwen3-8B												
Base Model	2.20	0.906	0.564	9.63	0.873	0.624	12.75	0.822	0.596	24.13	0.735	0.563
w/ **TTCL**	**3.50**	**0.352**	**0.623**	**14.81**	**0.190**	**0.752**	**15.31**	**0.330**	**0.798**	**26.50**	**0.226**	**0.799**
# Improv	+59.09%	+61.17%	+10.45%	+53.87%	+78.23%	+20.56%	+20.08%	+59.85%	+33.78%	+9.84%	+69.23%	+42.02%
Llama-3.1-8B-Instruct												
Base Model	3.20	0.812	0.535	14.69	0.670	0.516	12.31	0.714	0.514	36.88	0.451	0.522
w/ **TTCL**	**5.80**	**0.219**	**0.716**	**23.31**	**0.174**	**0.704**	**20.13**	**0.107**	**0.666**	**50.19**	**0.335**	**0.639**
# Improv	+81.25%	+72.98%	+33.82%	+40.23%	+73.83%	+35.36%	+64.29%	+85.10%	+23.79%	+30.78%	+25.00%	+24.08%

We next present two supporting results: the empirical probability $q(\mathrm{Ans}(y) \mid x)$ of the answer frequency can serve as a proxy for the latent conditional correctness probability $p^\star(x, y)$, while $p^\star(x, y)$ constitutes the ideal target for calibration learning.

Proposition 4.2 (**Answer Frequency as a Proxy for Conditional Correctness**). *Let $q_{\pi_\theta}(a \mid x)$ denote the population probability of answer a under the model. The empirical answer probability estimated from G independent responses is an unbiased and consistent estimator of $q_{\pi_\theta}(a \mid x)$. When answer distribution reflects model's uncertainty over the correct answer, $q_{\pi_\theta}(a \mid x)$ serves as a proxy for the probability that response y is correct, i.e., $p^\star(x, y)$, where $a = \mathrm{Ans}(y)$.*

Proposition 4.2 provides the statistical justification for using answer frequency as the label-free confidence target in TTCL. A detailed proof is provided in Appendix D.1.

We next show that the conditional correctness probability is the ideal target for calibration learning.

Proposition 4.3 (**Conditional Correctness as the Ideal Calibration Target**). *If the confidence estimator satisfies $c(X, Y) = p^\star(X, Y)$, then the resulting confidence is perfectly calibrated, i.e., $\Pr(Z = 1 \mid C = c) = c$. Minimizing the expected Brier score yields the optimal confidence estimator $c^\star = \arg\min_c \mathcal{J}_{\mathrm{Brier}}(c) = p^\star$.*

This proposition connects Brier score minimization to perfect calibration through the conditional correctness probability. A detailed proof is provided in Appendix D.2. Together with Proposition 4.2, this forms a theoretical bridge from answer frequency to conditional correctness probability and ultimately to calibrated confidence, supporting the surrogate guarantee in Theorem 4.1.

5 Experiments

5.1 Experimental Setup

Models. We conduct experiments on LLMs from different families and scales, including Qwen3-4B/8B (Yang et al., 2025) and Llama-3.1-8B-Instruct (Grattafiori et al., 2024), where Qwen3 is used in non-thinking mode. Besides applying TTCL directly to base models, we further evaluate it on models pre-calibrated with RLCR (Damani et al., 2026). Specifically, Qwen3-4B/8B are trained

Table 3: **In-domain performance of TTCL on RLCR-calibrated models.** Improved results are highlighted in **bold**, with positive and negative relative gains shown in red and green respectively.

Math-to-Math	MATH500			AMC			AIME24			AIME25			AIME26		
	ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑
RLCR-calibrated Qwen3-4B on DAPO-Math-14K															
Base Model	80.6	0.155	0.670	59.38	0.383	0.703	20.42	0.739	0.652	16.25	0.807	0.692	15.42	0.813	0.606
RLCR Model	89.0	0.158	0.867	79.38	0.131	0.865	29.17	0.291	0.881	24.17	0.311	0.888	23.33	0.348	0.823
w/ **TTCL**	**90.4**	**0.026**	**0.924**	**88.13**	**0.101**	**0.902**	**42.08**	**0.096**	**0.933**	**25.00**	**0.122**	**0.910**	**24.58**	**0.189**	**0.839**
# Improv	+1.57%	+83.37%	+6.67%	+11.02%	+22.76%	+4.31%	+44.29%	+67.18%	+5.93%	+3.45%	+60.57%	+2.52%	+5.36%	+45.52%	+2.02%
RLCR-calibrated Qwen3-8B on DAPO-Math-14K															
Base Model	84.4	0.183	0.717	62.81	0.338	0.651	21.67	0.753	0.642	18.75	0.789	0.620	13.75	0.839	0.597
RLCR Model	90.0	0.054	0.899	80.63	0.080	0.866	35.42	0.100	0.936	28.33	0.141	0.975	27.50	0.157	0.928
w/ **TTCL**	**91.6**	**0.042**	**0.967**	**89.38**	**0.065**	**0.881**	**45.42**	**0.074**	0.907	**30.42**	**0.127**	0.959	**28.33**	**0.122**	0.902
# Improv	+1.78%	+23.14%	+7.58%	+10.85%	+18.75%	+1.68%	+28.24%	+26.06%	-3.14%	+7.35%	+10.35%	-1.62%	+3.03%	+22.37%	-2.76%

Table 4: **Out-of-domain adaptation of TTCL on RLCR-calibrated models.** Improved results after TTCL are highlighted in **bold**, with positive relative improvements shown in red.

Math-to-FactQA	SimpleQA			ChineseSimpleQA			NQ			HotpotQA			TriviaQA		
	ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑
RLCR-calibrated Qwen3-4B on DAPO-Math-14K															
Base Model	1.40	0.912	0.481	12.13	0.819	0.658	8.63	0.859	0.705	9.56	0.831	0.703	15.63	0.789	0.660
RLCR Model	2.00	0.623	0.662	13.00	0.600	0.730	11.00	0.651	0.732	11.13	0.560	0.753	19.38	0.551	0.713
w/ **TTCL**	**2.70**	**0.277**	**0.747**	**14.03**	**0.307**	**0.798**	**12.81**	**0.228**	**0.867**	**13.50**	**0.293**	**0.850**	**20.38**	**0.290**	**0.812**
# Improv	+35.00%	+55.61%	+12.84%	+7.95%	+48.83%	+9.18%	+16.48%	+65.01%	+18.47%	+21.35%	+47.62%	+12.87%	+5.16%	+47.34%	+13.91%
RLCR-calibrated Qwen3-8B on DAPO-Math-14K															
Base Model	2.20	0.906	0.564	15.77	0.796	0.579	9.63	0.873	0.624	12.75	0.822	0.596	24.13	0.735	0.563
RLCR Model	2.80	0.768	0.591	18.17	0.640	0.676	12.56	0.730	0.658	14.13	0.633	0.696	25.69	0.599	0.691
w/ **TTCL**	**3.90**	**0.270**	**0.712**	**20.83**	**0.225**	**0.771**	**15.00**	**0.290**	**0.863**	**18.38**	**0.292**	**0.772**	**29.31**	**0.426**	**0.823**
# Improv	+39.29%	+64.85%	+20.46%	+14.68%	+64.92%	+14.07%	+19.40%	+60.30%	+31.12%	+30.09%	+53.85%	+10.90%	+14.11%	+28.98%	+19.15%
FactQA-to-Math	**MATH500**			**AMC**			**AIME24**			**AIME25**			**AIME26**		
	ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑
RLCR-calibrated Qwen3-4B on FactQA-10K															
Base Model	80.6	0.155	0.670	59.38	0.383	0.703	20.42	0.739	0.652	16.25	0.807	0.692	15.42	0.813	0.606
RLCR Model	79.8	0.114	0.818	64.38	0.103	0.779	23.75	0.286	0.814	18.75	0.322	0.822	14.58	0.334	0.772
w/ **TTCL**	**85.2**	**0.059**	**0.920**	**80.00**	**0.055**	**0.859**	**25.00**	**0.087**	**0.847**	**24.17**	**0.123**	**0.855**	**16.67**	**0.202**	0.743
# Improv	+6.77%	+48.33%	+12.48%	+24.27%	+46.37%	+10.22%	+5.26%	+69.46%	+3.96%	+28.89%	+61.68%	+3.97%	+14.29%	+39.44%	-3.69%

with RLCR for 3 epochs on DAPO-Math-14K and FactQA-10K, and the resulting models serve as calibrated starting points for evaluating TTCL adaptation to new target tasks.

Benchmarks. We evaluate TTCL on mathematical reasoning and factual QA benchmarks. The mathematics includes MATH500 (Lightman et al., 2024), AMC (Yang et al., 2024), AIME24 (Zhang & Math-AI, 2024), AIME25 (Zhang & Math-AI, 2025) and AIME26 (Zhang & Math-AI, 2026), while the factual QA includes SimpleQA (Wei et al., 2024), ChineseSimpleQA (He et al., 2025), NQ (Kwiatkowski et al., 2019), HotpotQA (Yang et al., 2018) and TriviaQA (Joshi et al., 2017). We apply TTCL to each benchmark independently without access its ground-truth labels, and evaluate the resulting model on the same benchmark. More details are provided in Appendix E.3.

Implementation details. We implement TTCL based on the VERL framework (Sheng et al., 2025). For each question, we sample $G = 32$ rollouts with a temperature of 0.7. We train TTCL with the learning rate of 5×10^{-7}. The target cache is updated after each epoch with an EMA rate of $\eta = 0.01$. Following prior studies (Damani et al., 2026), we evaluate answer correctness using Accuracy (ACC), calibration using Expected Calibration Error (ECE), and confidence discrimination using AUROC. More implementation and metric details refer to Appendix E.2 and Appendix E.4.

Baselines. Besides the corresponding starting models, we compare TTCL with several label-free calibration methods, including P(True) (Kadavath et al., 2022), Verbalization and Elicitation (Tian et al., 2023), and BaseCal (Tan et al., 2026). These methods mainly rely on post-hoc confidence estimation strategies to obtain more accurate confidence scores for given responses, without training the model. More details of the baselines are introduced in Appendix E.1.

5.2 Main Experimental Results

TTCL consistently improves both accuracy and calibration of base models. As shown in Table 1 and Table 2, we observe that TTCL consistently improves performance across mathematical reasoning and factual QA benchmarks. On mathematical reasoning, TTCL achieves average relative improvements of $+41.35\%$ in ACC and $+73.12\%$ in ECE; on factual QA, the corresponding gains are $+31.02\%$ and $+66.37\%$. These results suggest that TTCL substantially improves calibration while enhancing answer accuracy across diverse models and tasks. The AUROC gains further indicate improved confidence discrimination between correct and incorrect responses.

TTCL can further adapt calibrated models to new target tasks. From Tables 3 and 4, TTCL remains effective on RLCR-calibrated models. In the in-domain math-to-math setting, it further improves both accuracy and calibration. More importantly, RLCR calibration transfers imperfectly

Table 5: **Performance comparison with existing label-free calibration baselines.** The best results among label-free methods are highlighted in **bold**. RLCR is included as a label-leakage reference. AUROC is not reported for Llama-3.1-8B-Instruct on AIME25 because its accuracy is 0.

Methods	Label Leakage	MATH500			AMC			AIME24			AIME25			Average		
		ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑
Qwen3-4B																
RLCR	Yes	88.8	0.054	0.713	89.38	0.054	0.803	47.92	0.004	0.996	39.58	0.000	1.000	66.42	0.028	0.878
P(True)	No	80.6	0.155	0.786	59.38	0.315	0.697	20.42	0.214	0.865	16.25	0.306	0.701	44.16	0.248	0.762
Verbalization	No		0.266	0.602		0.326	0.706		0.485	0.750		0.636	0.543		0.428	0.650
Elicitation	No		0.096	0.754		0.281	0.718		0.486	0.800		0.592	0.779		0.364	0.763
BaseCal	No		0.062	0.598		0.270	0.618		0.636	0.675		0.667	0.717		0.409	0.652
TTCL	No	**86.6**	**0.029**	**0.866**	**77.81**	**0.091**	**0.910**	**28.33**	**0.065**	**0.856**	**20.42**	**0.261**	**0.878**	**53.29**	**0.112**	**0.878**
Qwen3-8B																
RLCR	Yes	91.2	0.070	0.727	95.94	0.028	0.807	61.67	0.074	0.959	47.08	0.095	0.994	73.97	0.067	0.872
P(True)	No	84.4	0.170	0.823	62.81	0.275	0.819	21.67	0.534	0.911	18.75	0.526	0.701	46.91	0.376	0.814
Verbalization	No		0.197	0.672		0.280	0.669		0.706	0.632		0.683	0.706		0.467	0.670
Elicitation	No		0.165	0.728		0.286	0.693		0.599	0.758		0.619	0.722		0.417	0.725
BaseCal	No		**0.096**	0.599		0.236	0.593		0.633	0.704		0.643	0.716		0.402	0.653
TTCL	No	**89.0**	0.131	**0.830**	**80.63**	**0.168**	**0.844**	**30.83**	**0.120**	**0.882**	**26.67**	**0.145**	**0.908**	**56.78**	**0.141**	**0.866**
Llama-3.1-8B-Instruct																
RLCR	Yes	71.0	0.100	0.688	82.19	0.109	0.791	39.58	0.040	0.994	0.00	0.010	-	48.19	0.065	0.824
P(True)	No	39.6	0.095	0.755	21.88	0.163	**0.620**	4.58	0.291	0.439	0.00	0.326	-	16.52	0.219	0.605
Verbalization	No		0.436	0.479		0.611	0.518		0.807	0.365		0.853	-		0.677	0.454
Elicitation	No		0.450	0.599		0.612	0.524		0.770	0.364		0.837	-		0.667	0.496
BaseCal	No		0.372	0.385		0.575	0.362		0.730	0.519		0.774	-		0.613	0.422
TTCL	No	**58.0**	**0.094**	**0.770**	**40.94**	**0.128**	0.617	**9.17**	**0.223**	**0.878**	0.00	**0.114**	-	**27.03**	**0.140**	**0.755**

Table 6: **Performance of TTCL models on other domains.** The results show that TTCL improves calibration without sacrificing task performance on other domains. More results of other TTCL models refer to Appendix F.5.

TTCL Target	ARC		GPQA-D	IFEval	MMLU-Pro	TruthfulQA	
	Challenge	Easy				MC1	MC2
Qwen3-8B							
Base Model	55.20	75.34	50.00	84.62	63.70	32.80	51.60
MATH500	55.29	74.83	51.52	85.47	64.82	33.29	51.65
AMC	55.29	74.83	45.96	85.83	63.87	33.05	51.35
AIME24	54.52	74.07	48.99	86.20	62.66	32.68	50.95
AIME25	54.52	74.12	51.01	84.98	63.16	32.80	50.82
RLCR-Qwen3-8B on DAPO-Math-14K							
RLCR Model	55.03	74.62	52.53	84.63	63.85	32.44	51.48
AMC	54.61	74.41	57.07	84.58	64.50	32.80	51.61
SimpleQA	55.03	74.03	52.53	84.28	64.07	32.93	50.49
ChineseSimpleQA	54.95	74.16	55.05	84.10	64.54	32.56	50.83
HotpotQA	54.86	74.33	52.53	85.38	64.40	32.93	51.12
TriviaQA	54.86	74.49	53.03	84.07	63.92	32.56	50.71

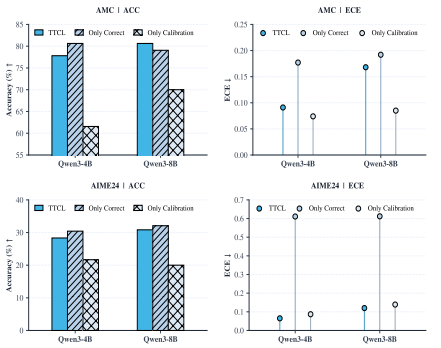


Figure 4: **Ablation study.** "Only Correct" and "Only Calibration" retain correctness and calibration rewards, respectively.

across domains: math-calibrated models remain poorly calibrated on factual QA tasks, and vice versa. In both math-to-FactQA and FactQA-to-math settings, TTCL consistently improves ACC and ECE, showing that it can refine calibration directly on unlabeled target-task data when transfer is limited.

TTCL consistently outperforms existing label-free calibration strategies. From Table 5, TTCL generally achieves higher ACC and AUROC and lower ECE than existing label-free baselines. As these methods mainly perform post-hoc confidence estimation without model training, their ACC remains unchanged, while TTCL further improves accuracy through target-task adaptation. RLCR performs strongly under label-leakage settings with access to ground-truth correctness labels, reflecting the benefit of explicit supervision. Notably, TTCL matches or surpasses RLCR in calibration in several cases without such labels, such as Llama-3.1-8B-Instruct on MATH500.

5.3 Further Analysis

TTCL improves calibration without sacrificing performance on other domains. Table 6 evaluates TTCL-trained models on ARC (Clark et al., 2018), GPQA-Diamond (Rein et al., 2023), IFEval (Zhou et al., 2023), MMLU-Pro (Wang et al., 2024), and TruthfulQA (Lin et al., 2022), covering scientific reasoning, instruction following, multi-domain knowledge, and truthfulness. TTCL-trained models maintain performance comparable to their pre-adaptation counterparts, indicating that TTCL improves target-task calibration while largely preserving general capabilities of the models.

Each component of TTCL contributes to its effectiveness. As shown in the ablation study in Fig. 4, optimizing only the correctness reward preserves accuracy but leads to substantially worse calibration, whereas optimizing only the calibration reward achieves low ECE at the cost of clear accuracy degradation. These results indicate that the two objectives are complementary, and combining them enables TTCL to achieve a better balance between accuracy and calibration.

Table 7: **Performance of binary pseudo-correctness as confidence supervision vs. TTCL confidence supervision.** Pseudo-RLCR uses the majority-vote pseudo-label for both correctness and confidence supervision signals, whereas TTCL uses answer frequency as the confidence supervision.

Model	Method	AMC23			AIME24			NQ			HotpotQA		
		ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑	ACC ↑	ECE ↓	AUROC ↑
Qwen3-4B	Pseudo-RLCR	74.38	0.158	0.698	26.67	0.595	0.574	**10.94**	0.836	0.688	**11.00**	0.853	0.554
	TTCL	**77.81**	**0.091**	**0.910**	**28.33**	**0.065**	**0.856**	10.63	**0.155**	**0.848**	10.50	**0.244**	**0.744**
Qwen3-8B	Pseudo-RLCR	72.50	0.249	0.735	**31.25**	0.450	0.744	**15.50**	0.751	0.553	14.00	0.847	0.553
	TTCL	**80.63**	**0.168**	**0.844**	30.83	**0.120**	**0.882**	14.81	**0.190**	**0.752**	**15.31**	**0.330**	**0.798**

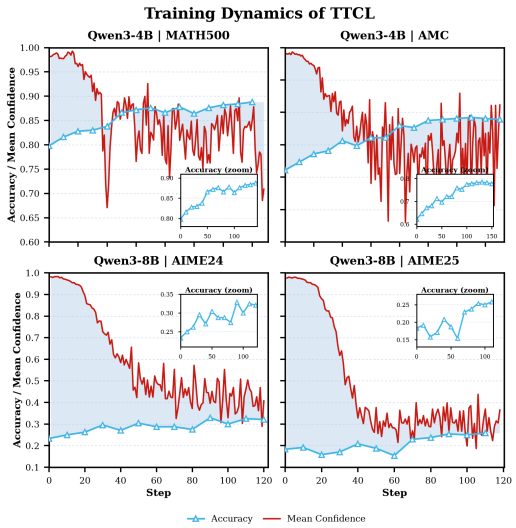


Figure 5: **Training dynamics of TTCL.** The dynamics illustrate the calibration process between accuracy and mean confidence during TTCL training. More dynamics refer to Appendix F.3.

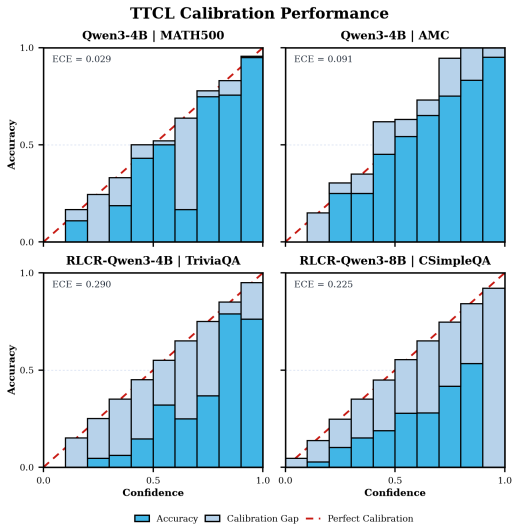


Figure 6: **Visulization of TTCL calibration performance across different settings.** Additional visualizations comparing calibration before and after TTCL are provided in Appendix F.2.

Frequency-based confidence supervision is more effective than binary pseudo-correctness supervision. Table 7 compares TTCL with Pseudo-RLCR, which uses the majority-vote pseudo-label for both correctness and confidence supervision. We observe that TTCL achieves lower ECE and higher ACC and AUROC in most cases, showing that binary pseudo-labels are insufficient for confidence learning and that frequency-based probability provides a more informative target.

TTCL progressively improves calibration during training. Fig. 5 shows the evolution of accuracy and mean confidence during TTCL training. The initial models are generally over-confident, with mean confidence exceeding empirical accuracy. As training proceeds, TTCL narrows this gap while improving accuracy, bringing the two increasingly into alignment. This training behavior empirically supports our theoretical interpretation of TTCL as a surrogate for the ideal calibration objective.

TTCL effectively aligns confidence with empirical accuracy. Beyond the scalar ECE metric, Fig. 6 visualizes confidence against empirical accuracy. Accuracy across confidence bins becomes closer to the ideal calibration diagonal after TTCL, indicating better alignment between predicted confidence and correctness likelihood. This pattern holds across mathematical reasoning and factual QA, as well as base and pre-calibrated models, showing that reliable confidence estimates of TTCL.

6 Conclusion

In this work, we introduce *Test-Time Calibration Learning (TTCL)*, a label-free framework for adapting LLM calibration directly to target tasks at test time. Unlike existing methods relying on labeled supervision, TTCL derives correctness and calibration self-supervision from multiple model-generated responses, enabling calibration learning without access to ground-truth labels. Theoretical analysis shows that TTCL provides a label-free surrogate for the ideal calibration objective. Extensive experiments on mathematical reasoning and factual QA demonstrate consistent improvements in accuracy and calibration. TTCL can further adapt pre-calibrated models under domain shift, improving calibration when source-domain transfer is limited. We hope this work inspires further research on test-time calibration for reliable confidence estimation in unlabeled deployment settings.

AI USE STATEMENT

In this work, we used generative AI tools to aid or polish writing, support research ideation and execution, and assist with proving mathematical claims. All AI-assisted outputs were carefully reviewed and validated by the authors. In particular, writing and research-related suggestions were independently assessed before use, and AI-assisted mathematical reasoning was checked for correctness by the authors. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work studies test-time calibration learning and does not involve human subjects, the collection of personally identifiable or sensitive information, or the release of new datasets. Our experiments are conducted on existing benchmarks and models for research purposes. We do not identify any specific ethical concerns beyond those generally associated with large language models. Nevertheless, calibrated confidence should not be interpreted as a guarantee of model correctness or safety, particularly in high-stakes applications. We have conducted this work in accordance with standard research integrity practices and the ICLR Code of Ethics.

REPRODUCIBILITY STATEMENT

We have made efforts to ensure the reproducibility of our results. The experimental setup section and appendix provide the details necessary to reproduce our experiments, including the training and evaluation configurations. We also release the source code and data at https://anonymous.4open.science/r/TTCL_Submission_code-ICLR27. We will further release all trained model checkpoints to facilitate independent reproduction and further evaluation of our method. Together, the experimental details, source code, data, and trained model weights provide the necessary resources for reproducing the results reported in this work.

REFERENCES

David Bani-Harouni, Chantal Pellegrini, Paul Stangel, Ege Özsoy, Kamilia Zaripova, Nassir Navab, and Matthias Keicher. Rewarding doubt: A reinforcement learning approach to calibrated confidence expression of large language models. In *ICLR*, 2026.

Michael Bereket and Jure Leskovec. Uncalibrated reasoning: Grpo induces overconfidence for stochastic outcomes. *arXiv preprint arXiv:2508.11800*, 2025.

Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 1950.

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 preprint arXiv:1803.05457*, 2018.

Mehul Damani, Isha Puri, Stewart Slocum, Idan Shenfeld, Leshem Choshen, Yoon Kim, and Jacob Andreas. Beyond binary rewards: Training lms to reason about their uncertainty. In *ICLR*, 2026.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. *Nature*, 2024.

Conor Finlay, Joshua Kurien, Saurabh Dash, Marzieh Fadaee, and Beyza Ermis. Caliber: Calibrating confidence before and after reasoning in language models. *arXiv preprint arXiv:2606.24281*, 2026.

Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 2007.

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.

Tobias Groot and Matias Valdenegro-Toro. Overconfidence is key: Verbalized uncertainty evaluation in large language and vision-language models. In *TrustNLP*, 2024.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *ICML*, 2017.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 2025.

Lukas Haas, Gal Yona, Giovanni D'Antonio, Sasha Goldshtein, and Dipanjan Das. Simpleqa verified: A reliable factuality benchmark to measure parametric knowledge. *arXiv preprint arXiv:2509.07968*, 2025.

Nathan Habib, Clémentine Fourrier, Hynek Kydlíček, Thomas Wolf, and Lewis Tunstall. Lighteval: A lightweight framework for llm evaluation, 2023. URL https://github.com/huggingface/lighteval.

Yancheng He, Shilong Li, Jiaheng Liu, Yingshui Tan, Weixun Wang, Hui Huang, Xingyuan Bu, Hangyu Guo, Chengwei Hu, Boren Zheng, et al. Chinese simpleqa: A chinese factuality evaluation for large language models. In *ACL*, 2025.

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. In *NeurIPS*, 2021.

Chaeyun Jang, Moonseok Choi, Yegon Kim, Seungyoo Lee, Juho Lee, and Hyungi Lee. Confidence is not universal: Task-dependent calibration and emergent behavior in llms. In *ICML*, 2026.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 2023.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *ACL*, 2017.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Adam Tauman Kalai, Ofir Nachum, Santosh S Vempala, and Edwin Zhang. Why language models hallucinate. *arXiv preprint arXiv:2509.04664*, 2025.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *ICLR*, 2023.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 2019.

Jixuan Leng, Chengsong Huang, Banghua Zhu, and Jiaxin Huang. Taming overconfidence in llms: Reward calibration in rlhf. In *ICLR*, 2025.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *ICLR*, 2024.

Stephanie Lin, Jacob Hilton, and Owain Evans. Truthfulqa: Measuring how models mimic human falsehoods. In *ACL*, 2022.

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.

Qing Lyu, Kumar Shridhar, Chaitanya Malaviya, Li Zhang, Yanai Elazar, Niket Tandon, Marianna Apidianaki, Mrinmaya Sachan, and Chris Callison-Burch. Calibrating large language models with sample consistency. In *AAAI*, 2025.

Zhengzhao Ma, Xueru Wen, Boxi Cao, Yaojie Lu, Hongyu Lin, Jinglin Yang, Min He, Xianpei Han, and Le Sun. Decoupling reasoning and confidence: Resurrecting calibration in reinforcement learning from verifiable rewards. In *ICML*, 2026.

Andrey Malinin and Mark Gales. Uncertainty estimation in autoregressive structured prediction. *arXiv preprint arXiv:2002.07650*, 2020.

OpenAI. Learning to reason with llms. https://openai.com/index/learning-to-reason-with-llms/, 2024.

Hadas Orgad, Michael Toker, Zorik Gekhman, Roi Reichart, Idan Szpektor, Hadas Kotek, and Yonatan Belinkov. Llms know more than they show: On the intrinsic representation of llm hallucinations. In *International Conference on Learning Representations*, volume 2025, pp. 66880–66913, 2025.

Mihir Prabhudesai, Lili Chen, Alex Ippoliti, Katerina Fragkiadaki, Hao Liu, and Deepak Pathak. Maximizing confidence alone improves reasoning. *arXiv preprint arXiv:2505.22660*, 2025.

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. *arXiv preprint arXiv:2311.12022*, 2023.

KiJung Seo, Sehun Lim, and Taeuk Kim. Advice: Answer-dependent verbalized confidence estimation. In *ACL*, 2026.

Sheikh Shafayat, Fahim Tajwar, Ruslan Salakhutdinov, Jeff Schneider, and Andrea Zanette. Can large reasoning models self-train? *arXiv preprint arXiv:2505.21444*, 2025.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. In *EuroSys*, 2025.

Jingwei Sun, Jianing Zhu, Jiangchao Yao, Tongliang Liu, and Bo Han. Rethinking how to remember: Beyond atomic facts in lifelong llm agent memory, 2026. URL https://arxiv.org/abs/2605.19952.

Hexiang Tan, Wanli Yang, Junwei Zhang, Xin Chen, Rui Tang, Du Su, Jingang Wang, Yuanzhuo Wang, Fei Sun, and Xueqi Cheng. BaseCal: Unsupervised confidence calibration via base model signals. In *ACL*, 2026.

Kimi Team, Tongtong Bai, Yifan Bai, Yiping Bao, Jianfeng Cai, Xinyuan Cai, Peizhou Cao, Yuxuan Cao, Ziwei Chai, Y Charles, et al. Kimi k3: Open frontier intelligence. *arXiv preprint arXiv:2607.24653*, 2026.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher Manning. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In *EMNLP*, 2023.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *ICLR*, 2023.

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. In *NeurIPS*, 2024.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *NeurIPS*, 2022.

Jason Wei, Nguyen Karina, Hyung Won Chung, Yunxin Joy Jiao, Spencer Papay, Amelia Glaese, John Schulman, and William Fedus. Measuring short-form factuality in large language models. *arXiv preprint arXiv:2411.04368*, 2024.

Xuexiang Wen, Hang Yu, Linchao Zhu, and Gaoang Wang. Verifier-free rl for llms via intrinsic gradient-norm reward. In *Findings of ACL*, 2026.

Jiayun Wu, Jiashuo Liu, Zhiyuan Zeng, Tianyang Zhan, Tianle Cai, and Wenhao Huang. Mitigating llm hallucination via behaviorally calibrated reinforcement learning. *arXiv preprint arXiv:2512.19920*, 2025.

Bangjun Xiao, Bingquan Xia, Bo Yang, Bofei Gao, Bowen Shen, Chen Zhang, Chenhong He, Chiheng Lou, Fuli Luo, Gang Wang, et al. Mimo-v2-flash technical report. *arXiv preprint arXiv:2601.02780*, 2026.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, et al. Qwen2.5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122*, 2024.

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*, 2025.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *EMNLP*, 2018.

Zhiqin Yang, Jingwen Fu, Yuhan Liu, Hengyu Liu, Yonggang Zhang, Kainan Cao, Zizhuo Zhang, Chenxin Li, Ruibin Yuan, Jiahao Pan, et al. Scaling large reasoning models beyond human supervision: A path toward superintelligence. *arXiv preprint arXiv:2608.31075*, 2026.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In *NeurIPS*, 2023.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, juncai liu, LingJun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Ru Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Yonghui Wu, and Mingxuan Wang. Dapo: An open-source llm reinforcement learning system at scale. In *NeurIPS*, 2025.

Kongcheng Zhang, Qi Yao, Shunyu Liu, Yingjie Wang, Baisheng Lai, Jieping Ye, Mingli Song, and Dacheng Tao. Consistent paths lead to truth: Self-rewarding reinforcement learning for llm reasoning. In *NeurIPS*, 2025a.

Qingyang Zhang, Haitao Wu, Changqing Zhang, Peilin Zhao, and Yatao Bian. Right question is already half the answer: Fully unsupervised llm reasoning incentivization. In *NeurIPS*, 2025b.

Yanzhi Zhang, Zhaoxi Zhang, Haoxiang Guan, Yilin Cheng, Yitong Duan, Chen Wang, Yue Wang, Shuxin Zheng, and Jiyan He. No free lunch: Rethinking internal feedback for llm reasoning. *arXiv preprint arXiv:2506.17219*, 2025c.

Yifan Zhang and Team Math-AI. American invitational mathematics examination (aime) 2024, 2024.

Yifan Zhang and Team Math-AI. American invitational mathematics examination (aime) 2025, 2025.

Yifan Zhang and Team Math-AI. American invitational mathematics examination (aime) 2026, 2026.

Zizhuo Zhang, Jianing ZHU, Xinmu Ge, Zihua Zhao, (Andrew) Zhanke Zhou, Xuan Li, Xiao Feng, Jiangchao Yao, and Bo Han. Co-rewarding: Stable self-supervised rl for eliciting reasoning in large language models. In *ICLR*, 2026.

Xuandong Zhao, Zhewei Kang, Aosong Feng, Sergey Levine, and Dawn Song. Learning to reason without external rewards. In *ICLR*, 2026.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*, 2023.

Thomas Zollo, Jimmy Wang, and Richard Zemel. Unsupervised confidence calibration for reasoning llms from a single generation. *arXiv preprint arXiv:2604.19444*, 2026.

Yuxin Zuo, Kaiyan Zhang, Li Sheng, Shang Qu, Ganqu Cui, Xuekai Zhu, Haozhan Li, Xinwei Long, Ermo Hua, Biqing Qi, et al. Ttrl: Test-time reinforcement learning. In *NeurIPS*, 2025.

A GROUP RELATIVE POLICY OPTIMIZATION (GRPO)

Group Relative Policy Optimization (GRPO). GRPO is a widely used RL algorithm for LLM reasoning. Given a question x, GRPO samples a group of G responses $\{y_i\}_{i=1}^{G}$, each associated with a reward r_i, and optimizes the policy LLM using the group-relative advantages (Shao et al., 2024):

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}\left[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|y_i|}\sum_{t=1}^{|y_i|}\left(\rho_{i,t}(\theta)\hat{A}_i\right)\right], \quad \hat{A}_i = r_i - \text{Mean}\left(\{r_i\}_{i=1}^{G}\right), \tag{13}$$

where $\rho_{i,t}(\theta)$ denotes the PPO-style clipped importance ratio between the current policy π_θ and the rollout policy $\pi_{\theta_{\text{old}}}$. Different from standard GRPO, we use mean-only advantage normalization without dividing by the reward standard deviation. As the Brier calibration reward is continuous, standard-deviation normalization can amplify small reward differences and lead to unstable or extreme calibration updates (Bereket & Leskovec, 2025). While several improved variants of GRPO have been proposed, such as DAPO (Yu et al., 2025) and Dr.GRPO (Liu et al., 2025), our work primarily focuses on test-time calibration learning of confidence estimation. We adopt GRPO as our policy optimization algorithm throughout our experiments.

B RELATED WORK

In this section, we mainly review related work from three perspectives: LLM reasoning, label-free RL methods for LLM reasoning, and existing LLM calibration methods, to better position TTCL within the broader research landscape and demonstrate its distinctions from prior studies.

Large language model (LLM) reasoning. The reasoning capabilities of LLMs have advanced rapidly in recent years. Early studies primarily elicit reasoning capability through prompting and inference-time strategies, such as Chain-of-Thought (CoT) prompting (Wei et al., 2022), Tree-of-Thoughts (ToT) (Yao et al., 2023), or self-consistency (Wang et al., 2023). More recently, reinforcement learning (RL) has emerged as an effective paradigm for further eliciting and strengthening LLM reasoning capabilities (Shao et al., 2024; OpenAI, 2024), with reinforcement learning with verifiable rewards (RLVR) becoming a widely adopted approach in modern LLM post-training pipelines (Guo et al., 2025; Team et al., 2026; Xiao et al., 2026; Yang et al., 2025). However, improvements in reasoning accuracy do not necessarily translate into reliable uncertainty estimation. Existing studies have shown that LLMs can exhibit substantial over-confidence (Groot & Valdenegro-Toro, 2024; Leng et al., 2025), while RL training that mainly rewards answer correctness may further exacerbate such miscalibration (Kalai et al., 2025; Bereket & Leskovec, 2025). This poses substantial risks for deploying LLMs in high-risk scenarios, where confidently incorrect predictions may lead to severe consequences. Therefore, beyond improving reasoning accuracy, reliable reasoning models should also faithfully reflect the uncertainty of their predictions through well-calibrated confidence estimates.

Label-free RL methods for LLM reasoning. Recently, emerging studies have explored RL without annotated ground-truth answers to improve LLM reasoning capabilities, reducing reliance on labeled supervision (Yang et al., 2026). According to the source of self-supervised signals, these methods can be broadly divided into two categories. One line of work mainly encourages self-consistency among model outputs (Zuo et al., 2025; Zhang et al., 2025a), for example, using majority-voted answers as pseudo labels to supervise RL optimization (Zhang et al., 2026; Shafayat et al., 2025). Another line constructs rewards from the model's internal signals (Zhang et al., 2025c; Wen et al., 2026), such as minimizing prediction entropy (Zhang et al., 2025b; Prabhudesai et al., 2025) or maximizing self-certainty (Zhao et al., 2026), to sharpen the model's predictive distribution. Although these methods enable label-free improvement of reasoning performance, they primarily focus on answer correctness and lack an explicit optimization objective for calibration learning. As a result, the resulting models may still fail to express confidence that faithfully reflects uncertainty over their own predictions. In contrast, our work further explores label-free learning for LLM calibration, enabling models to adapt both reasoning capability and confidence estimation on target tasks at test time.

Existing LLM calibration methods. Existing approaches to LLM calibration can be broadly categorized into post-hoc calibration and calibration learning. Post-hoc methods estimate confidence for model-generated responses using signals from multiple perspectives, such as token probabilities (Tan et al., 2026), verbalized confidence (Tian et al., 2023), self-evaluation (Kadavath et al., 2022), sampling consistency (Lyu et al., 2025), or semantic uncertainty (Farquhar et al., 2024; Kuhn et al., 2023).

While effective in many settings, these methods primarily derive confidence through external signals or post-processing, rather than explicitly training the LLM to develop the capability of estimating the confidence of its own generated responses. In contrast, recent calibration learning studies directly train LLMs to express their calibrated confidence, typically by incorporating calibration-aware objectives into existing LLM training objectives (Ma et al., 2026; Seo et al., 2026; Finlay et al., 2026). For example, RLCR (Damani et al., 2026), BCRL (Wu et al., 2025), and Rewarding Doubt (Bani-Harouni et al., 2026) leverage proper scoring rules (Gneiting & Raftery, 2007) to introduce calibration rewards into RL training, jointly optimizing answer correctness and confidence estimation.

These methods explicitly encourage calibration to become an intrinsic capability of the model. However, their calibration supervision generally relies on ground-truth correctness labels during training (Tan et al., 2026), incurring additional annotation costs, while the learned calibration behavior may transfer poorly when the target distribution differs from the training distribution (Jang et al., 2026). In this work, we instead introduce *Test-Time Calibration Learning (TTCL)*, which directly adapts model calibration on the target task at test time using label-free supervision without requiring ground-truth labels, thereby enabling calibration learning under unseen test distributions.

C Pseudocode of TTCL

To provide an intuitive overview of the TTCL pipeline, we summarize its pseudocode in Algorithm 1. Specifically, the target cache is initialized once using the initial policy model and remains fixed throughout each epoch. It provides the pseudo-correctness and confidence targets and is updated only at the end of each epoch using the newly observed answer distribution.

D Further Theoretical Analysis of TTCL

We provide the detailed proofs for the theoretical analysis presented in Section 4. We first establish the statistical property of the frequency-based probability confidence target, then show that the conditional correctness probability is the ideal target for calibration learning, and finally prove the main analysis that the TTCL objective provides a bounded surrogate for the ideal calibration objective.

D.1 Proof of Proposition 4.2

Proof of **Proposition 4.2 (Answer Frequency as a Proxy for Conditional Correctness)**.

Proof. For a fixed question x and candidate answer a, consider G sampled responses

$$\{Y_1, \dots, Y_G\} \sim \pi_\theta(\cdot \mid x), \tag{14}$$

and let $A_j = \mathrm{Ans}(Y_j)$ denote the extracted answer of the j-th response. Define the empirical answer probability as

$$\hat{q}_G(a \mid x) = \frac{1}{G} \sum_{j=1}^{G} \mathbb{I}[A_j \equiv a]. \tag{15}$$

Since

$$\mathbb{E}[\mathbb{I}[A_j \equiv a]] = \Pr(A_j = a \mid X = x) = q_{\pi_\theta}(a \mid x), \tag{16}$$

We obtain

$$\mathbb{E}[\hat{q}_G(a \mid x)] = \frac{1}{G} \sum_{j=1}^{G} q_{\pi_\theta}(a \mid x) = q_{\pi_\theta}(a \mid x). \tag{17}$$

Therefore, $\hat{q}_G(a \mid x)$ is an unbiased estimator of the population answer probability. Moreover, each indicator variable is Bernoulli distributed with parameter $q_{\pi_\theta}(a \mid x)$. Hence,

$$\mathrm{Var}[\hat{q}_G(a \mid x)] = \frac{q_{\pi_\theta}(a \mid x)\,(1 - q_{\pi_\theta}(a \mid x))}{G}. \tag{18}$$

As $G \to \infty$,

$$\mathrm{Var}[\hat{q}_G(a \mid x)] \to 0, \tag{19}$$

Algorithm 1 Test-Time Calibration Learning (TTCL)

Input: Unlabeled target set $\mathcal{D}_{\text{test}}$; initial policy model π_{θ_0}; number of training epochs E; learning rate α; cache update rate η.
Output: Calibration-adapted policy model π_θ
Initialize $\theta \leftarrow \theta_0$
Initialize an empty target cache $q_{\text{cache}}^{(0)}$
for all $x \in \mathcal{D}_{\text{test}}$ **do**
    Sample G responses from $\pi_{\theta_0}(\cdot \mid x)$
    Set $q_{\text{cache}}^{(0)}(\cdot \mid x)$ to their empirical answer distribution
end for
for $e = 0, \ldots, E-1$ **do**
    Freeze $q_{\text{cache}}^{(e)}$ for supervision in epoch e
    for all minibatches $\mathcal{B} \subset \mathcal{D}_{\text{test}}$ **do**
        for all input question $x \in \mathcal{B}$ **do**
            Sample $\{y_i = (\tau_i, a_i, c_i)\}_{i=1}^{G}$ from the rollout policy $\pi_{\theta_{\text{old}}}(\cdot \mid x)$.
            $\tilde{a}_x^{(e)} \leftarrow \arg\max_a q_{\text{cache}}^{(e)}(a \mid x)$. # *Read the pseudo-correctness target from the cache*
            for $i = 1, \ldots, G$ **do**
                $\tilde{p}_i^{(e)} \leftarrow q_{\text{cache}}^{(e)}(a_i \mid x)$. # *Read the confidence target from the cache*
                $r_i^{\text{corr}} \leftarrow \mathbb{I}[a_i \equiv \tilde{a}_x^{(e)}]$. # *Compute correctness reward*
                $r_i^{\text{cal}} \leftarrow -(c_i - \tilde{p}_i^{(e)})^2$. # *Compute calibration reward*
            end for
            Compute mean-centered advantages $\{\hat{A}_i^{\text{corr}}, \hat{A}_i^{\text{cal}}\}_{i=1}^{G}$.
        end for
        Compute TTCL objective in Eq. (9).
        Update $\theta \leftarrow \theta + \alpha \nabla_\theta \mathcal{J}_{\text{TTCL}}(\theta)$.
    end for
    # *EMA updates the target cache after each epoch*
    for all $x \in \mathcal{D}_{\text{test}}$ **do**
        Let $q^{(e)}(\cdot \mid x)$ be the empirical answer distribution observed in epoch e.
        Update $q_{\text{cache}}^{(e+1)}(\cdot \mid x) \leftarrow (1-\eta) q_{\text{cache}}^{(e)}(\cdot \mid x) + \eta q^{(e)}(\cdot \mid x)$ using EMA.
    end for
end for
return Calibrated policy model π_θ

and by the law of large numbers,

$$\hat{q}_G(a \mid x) \xrightarrow{p} q_{\pi_\theta}(a \mid x). \tag{20}$$

Thus, the empirical answer probability is both unbiased and consistent.

For a response y whose extracted answer is $a = \text{Ans}(y)$, when the model-induced answer distribution reflects the model's uncertainty over which answer is correct, answers with larger $q_{\pi_\theta}(a \mid x)$ correspond to larger probabilities of the associated response being correct. Therefore,

$$q_{\pi_\theta}(a \mid x) \tag{21}$$

provides a frequency-based proxy for

$$p^\star(x, y) = \Pr(Z = 1 \mid X = x, Y = y). \tag{22}$$

Since $\hat{q}_G(a \mid x)$ consistently estimates $q_{\pi_\theta}(a \mid x)$, repeated responses provide a statistically well-defined way to estimate this proxy without access to ground-truth correctness labels. □

D.2 Proof of Proposition 4.3

Proof of **Proposition 4.3 (Conditional Correctness as the Ideal Calibration Target)**.

Proof. We first establish the connection between conditional correctness probability and perfect calibration. By definition,

$$p^\star(X, Y) = \Pr(Z = 1 \mid X, Y). \tag{23}$$

Since $Z \in \{0, 1\}$,

$$\mathbb{E}[Z \mid X, Y] = p^\star(X, Y). \tag{24}$$

Suppose the confidence estimator satisfies

$$C = c(X, Y) = p^\star(X, Y). \tag{25}$$

Using the law of iterated expectations,

$$\mathbb{E}[Z \mid C] = \mathbb{E}[\mathbb{E}[Z \mid X, Y] \mid C]\,. \tag{26}$$

Substituting the definition of $p^\star$,

$$\mathbb{E}[Z \mid C] = \mathbb{E}[p^\star(X, Y) \mid C]\,. \tag{27}$$

Since $C = p^\star(X, Y)$,

$$\mathbb{E}[Z \mid C] = C. \tag{28}$$

Conditioning on $C = c$ therefore gives

$$\Pr(Z = 1 \mid C = c) = \mathbb{E}[Z \mid C = c] = c, \tag{29}$$

which is exactly the perfect calibration condition in Eq. (1).

We next show that the expected Brier score recovers the same target. Consider

$$\mathcal{J}_{\text{Brier}}(c) = \mathbb{E}\big[(c(X, Y) - Z)^2\big]\,. \tag{30}$$

Conditioning on (X, Y), and writing $c = c(X, Y)$ and $p^\star = p^\star(X, Y)$ for simplicity, we have

$$\mathbb{E}\big[(c - Z)^2 \mid X, Y\big]\,. \tag{31}$$

Adding and subtracting $p^\star$,

$$c - Z = (c - p^\star) + (p^\star - Z). \tag{32}$$

Therefore,

$$\mathbb{E}\big[(c - Z)^2 \mid X, Y\big] = (c - p^\star)^2 + 2(c - p^\star)\,\mathbb{E}[p^\star - Z \mid X, Y] + \mathbb{E}\big[(p^\star - Z)^2 \mid X, Y\big]\,. \tag{33}$$

Since

$$\mathbb{E}[Z \mid X, Y] = p^\star, \tag{34}$$

the cross term vanishes:

$$\mathbb{E}[p^\star - Z \mid X, Y] = 0. \tag{35}$$

Hence,

$$\mathbb{E}\big[(c - Z)^2 \mid X, Y\big] = (c - p^\star)^2 + \mathbb{E}\big[(p^\star - Z)^2 \mid X, Y\big]\,. \tag{36}$$

Because Z is Bernoulli with conditional mean $p^\star$,

$$\mathbb{E}\big[(p^\star - Z)^2 \mid X, Y\big] = p^\star(1 - p^\star). \tag{37}$$

Taking expectation over (X, Y) yields

$$\mathcal{J}_{\text{Brier}}(c) = \mathbb{E}\big[(c(X, Y) - p^\star(X, Y))^2\big] + \mathbb{E}[p^\star(X, Y)(1 - p^\star(X, Y))]\,. \tag{38}$$

The second term is independent of the confidence estimator c. Therefore, minimizing the expected Brier score is equivalent to minimizing

$$\mathbb{E}\big[(c(X, Y) - p^\star(X, Y))^2\big]\,, \tag{39}$$

whose minimum is attained at

$$c^\star(X, Y) = p^\star(X, Y). \tag{40}$$

Thus,

$$c^\star = \arg\min_c \mathcal{J}_{\text{Brier}}(c) = p^\star. \tag{41}$$

Therefore, the conditional correctness probability is both sufficient for perfect calibration and the optimal target under Brier-score minimization. □

D.3 Proof of Theorem 4.1

With the frequency-based proxy and the ideal calibration target established above, we now prove the surrogate bound in Theorem 4.1.

Proof of **Theorem 4.1 (TTCL as a Surrogate for the Ideal Calibration Objective)**.

Proof. Recall that

$$\mathcal{J}_{\mathrm{TTCL}}(c) = \mathbb{E}\left[(c(X,Y) - s(X,Y))^2\right], \tag{42}$$

and

$$\mathcal{J}_{\mathrm{ideal}}(c) = \mathbb{E}\left[(c(X,Y) - p^\star(X,Y))^2\right]. \tag{43}$$

The estimation error is defined as

$$\epsilon = \mathbb{E}[|s(X,Y) - p^\star(X,Y)|]. \tag{44}$$

For notational simplicity, in this proof, let

$$c = c(X,Y), \qquad s = s(X,Y), \qquad p^\star = p^\star(X,Y). \tag{45}$$

We first consider the pointwise difference between the two squared-error objectives:

$$\left|(c-s)^2 - (c-p^\star)^2\right|. \tag{46}$$

Using the difference-of-squares identity,

$$\left|(c-s)^2 - (c-p^\star)^2\right| = |(p^\star - s)(2c - s - p^\star)| \tag{47}$$

$$= |s - p^\star|\,|2c - s - p^\star|. \tag{48}$$

Since

$$c, s, p^\star \in [0,1], \tag{49}$$

we have

$$|2c - s - p^\star| \le 2. \tag{50}$$

Therefore,

$$\left|(c-s)^2 - (c-p^\star)^2\right| \le 2|s - p^\star|. \tag{51}$$

Taking expectation on both sides and applying

$$|\mathbb{E}[U]| \le \mathbb{E}[|U|], \tag{52}$$

we obtain

$$|\mathcal{J}_{\mathrm{TTCL}}(c) - \mathcal{J}_{\mathrm{ideal}}(c)| = \left|\mathbb{E}\left[(c-s)^2 - (c-p^\star)^2\right]\right| \tag{53}$$

$$\le \mathbb{E}\left[\left|(c-s)^2 - (c-p^\star)^2\right|\right] \tag{54}$$

$$\le 2\mathbb{E}[|s - p^\star|] \tag{55}$$

$$= 2\epsilon. \tag{56}$$

Therefore,

$$|\mathcal{J}_{\mathrm{TTCL}}(c) - \mathcal{J}_{\mathrm{ideal}}(c)| \le 2\epsilon, \tag{57}$$

for any confidence estimator $c(X,Y) \in [0,1]$. This establishes that TTCL provides a bounded surrogate for the ideal calibration learning objective. □

E Additional Experimental Details

E.1 Details of Baselines

We compare TTCL with several existing label-free calibration strategies and additionally include RLCR, a calibration learning method with access to ground-truth labels, as a label-leakage reference. We briefly introduce these methods below:

- **RLCR** (Damani et al., 2026): Reinforcement Learning with Calibration Rewards which jointly optimizes answer correctness and verbalized confidence by augmenting the binary correctness reward with a Brier-score-based calibration reward. Unlike TTCL, RLCR requires ground-truth answer labels to construct its training reward for both correctness and calibration.

Table 8: **Detailed implementation configurations.**

Setting	MATH500	AMC	AIME24/AIME25/AIME26	SimpleQA	ChineseSimpleQA	NQ/HotpotQA/TriviaQA
Training epochs	10	40	40	10	10	10
Training batch size	32	8	8	64	128	32
Training Temperature	0.7					
Rollouts G	32					
Max prompt length	1,024					
Max response length	4,096					
Learning rate	5e-7					
KL coefficient	0.001					
Optimizer	AdamW ($\beta_1 = 0.9, \beta_2 = 0.999$)					
Clip ratio	0.2					
Warmup style	Cosine					
Warmup step ratio	0.1					
Evaluation Temperature	0.7					

Table 9: **Statistics of benchmarks** used in our experiments.

Task	Dataset Name	# Data Size
Mathematics	MATH500 (Lightman et al., 2024)	500
	AMC (Yang et al., 2024)	40
	AIME24 (Zhang & Math-AI, 2024)	30
	AIME25 (Zhang & Math-AI, 2025)	30
	AIME26 (Zhang & Math-AI, 2026)	30
	DAPO-Math-14K (Yu et al., 2025)	14,109
Factual QA	SimpleQA (Wei et al., 2024; Haas et al., 2025)	1,000
	ChineseSimpleQA (He et al., 2025)	3,000
	NQ (Kwiatkowski et al., 2019)	200
	HotpotQA (Yang et al., 2018)	200
	TriviaQA (Joshi et al., 2017)	200
	FactQA-10K (Yang et al., 2018; Joshi et al., 2017)	10,000

- **P(True)** (Kadavath et al., 2022): P(True) estimates confidence through self-evaluation. Given a question and the generated answer, the model is prompted to judge whether the proposed answer is true or false, and the predicted probability assigned to "True" is used as the confidence score.
- **Verbalization** (Tian et al., 2023): Verbalization directly prompts the model to report a numerical confidence together with its answer, using the model's instruction-following ability to express uncertainty without accessing internal logits or additional labeled data.
- **Elicitation** (Tian et al., 2023): Elicitation improves confidence verbalization by obtaining multiple confidence estimates through repeated prompting and sampling, and aggregating these estimates to produce the final confidence score.
- **BaseCal** (Tan et al., 2026): BaseCal exploits the observation that the corresponding pre-trained base model can provide better-calibrated signals than its post-trained counterpart. Following BaseCal-ReEval, we re-evaluate the response generated by the post-trained model using the base model and aggregate the token probabilities as the calibrated confidence.

For a fair comparison, we use the same evaluation settings for all methods.

E.2 More Implementation Details

For both training and evaluation, we use the same system prompt as follows:

System prompt for mathematical reasoning

Let's think step by step and output the final answer within \boxed{}.
Also output your confidence for this answer after <conf> as a floating−point number between 0 (complete uncertainty) and 1 (full confidence).
Example: <conf> Confidence: <float value>
Ensure that the confidence output appears on a single line and strictly follows the specified format.

Table 10: **Brier score and mean confidence of TTCL for base models on mathematical reasoning benchmarks.** Improved results after TTCL are highlighted in **bold**, with positive relative improvements shown in red. As mean confidence is not a metric for which higher or lower values are better, we do not report its relative improvement, and it only serves as a reference alongside other metrics.

Methods	MATH500			AMC			AIME24			AIME25		
	ACC ↑	Brier ↓	Conf	ACC ↑	Brier ↓	Conf	ACC ↑	Brier ↓	Conf	ACC ↑	Brier ↓	Conf
Qwen3-4B												
Base Model	80.6	0.184	0.984	59.38	0.387	0.984	20.42	0.731	0.951	16.25	0.790	0.979
w/ **TTCL**	**86.6**	**0.071**	0.881	**77.81**	**0.131**	0.709	**28.33**	**0.144**	0.325	**20.42**	**0.185**	0.470
# Improv	+7.44%	+61.48%	-	+31.05%	+66.15%	-	+38.78%	+80.29%	-	+25.64%	+76.58%	-
Qwen3-8B												
Base Model	84.4	0.219	0.987	62.81	0.357	0.983	21.67	0.757	0.978	18.75	0.780	0.981
w/ **TTCL**	**89.0**	**0.090**	0.769	**80.63**	**0.145**	0.844	**30.83**	**0.177**	0.392	**26.67**	**0.113**	0.303
# Improv	+5.45%	+59.17%	-	+28.36%	+59.30%	-	+42.31%	+76.63%	-	+42.22%	+85.53%	-
Llama-3.1-8B-Instruct												
Base Model	39.6	0.480	0.858	21.88	0.605	0.518	4.58	0.731	0.856	0.00	0.751	0.853
w/ **TTCL**	**58.0**	**0.208**	0.622	**40.94**	**0.261**	0.617	**9.17**	**0.112**	0.284	0.00	**0.036**	0.114
# Improv	+46.46%	+56.68%	-	+87.14%	+56.80%	-	+100.00%	+84.69%	-	0.00%	+95.25%	-

Table 11: **Brier score and mean confidence of TTCL for base models on factual QA benchmarks.** Improved results after TTCL are highlighted in **bold**, with positive relative improvements shown in red color. As mean confidence is not a metric for which higher or lower values are better, we do not report its relative improvement and include it only as a reference alongside the other metrics.

Methods	SimpleQA			NQ			HotpotQA			TriviaQA		
	ACC ↑	Brier ↓	Conf	ACC ↑	Brier ↓	Conf	ACC ↑	Brier ↓	Conf	ACC ↑	Brier ↓	Conf
Qwen3-4B												
Base Model	1.40	0.875	0.927	8.63	0.855	0.961	9.56	0.795	0.927	15.63	0.772	0.954
w/ **TTCL**	**2.40**	**0.041**	0.095	**10.63**	**0.176**	0.420	**10.06**	**0.198**	0.349	**17.00**	**0.099**	0.191
# Improv	+71.43%	+95.36%	-	+23.25%	+79.41%	-	+5.20%	+75.06%	-	+8.80%	+87.22%	-
Qwen3-8B												
Base Model	2.20	0.886	0.928	9.63	0.876	0.983	12.75	0.810	0.950	24.13	0.743	0.989
w/ **TTCL**	**3.50**	**0.184**	0.387	**14.81**	**0.154**	0.338	**15.31**	**0.240**	0.483	**26.50**	**0.216**	0.492
# Improv	+59.09%	+79.29%	-	+53.87%	+82.39%	-	+20.08%	+70.36%	-	+9.84%	+70.91%	-
Llama-3.1-8B-Instruct												
Base Model	3.20	0.719	0.846	14.69	0.666	0.854	12.31	0.662	0.851	36.88	0.479	0.857
w/ **TTCL**	**5.80**	**0.111**	0.278	**23.31**	**0.203**	0.065	**20.13**	**0.163**	0.098	**50.19**	**0.352**	0.169
# Improv	+81.25%	+84.51%	-	+40.23%	+69.55%	-	+64.29%	+75.34%	-	+30.78%	+26.44%	-

System prompt for factual question answering

Let's think step by step and output the final answer within <answer> </answer> tags.
Also output your confidence for this answer after <conf> as a floating−point number between 0 (complete uncertainty) and 1 (full confidence).
Example: <conf> Confidence: <float value>
Ensure that the confidence output appears on a single line and strictly follows the specified format.

The first prompt is used for mathematical reasoning, while the second is used for factual question answering; they differ only in the requirement for final answer extraction.

Table 8 summarizes the implementation details of TTCL for each benchmark. Following a similar principle adopted in existing test-time training methods (Zuo et al., 2025), we use smaller batch sizes and more training epochs for smaller benchmarks to ensure a sufficient number of training steps. Additionally, all our experiments are conducted on 8× Ascend 910B NPUs.

E.3 More Benchmark Details

We conduct experiments on benchmarks mainly covering two tasks: mathematical reasoning and factual question answering. For mathematical reasoning, we include benchmarks below:

- **MATH500** (Lightman et al., 2024): MATH500 is a commonly used benchmark for mathematics, which is a 500-question subset of the MATH dataset (Hendrycks et al., 2021), covering a broad range of topics including algebra, geometry, number theory, counting, and probability.
- **AMC** (Yang et al., 2024): AMC contains problems drawn from the American Mathematics Competitions, which evaluate math problem-solving ability across topics such as algebra, geometry,

Table 12: **In-domain performance of Brier score and mean confidence of TTCL on RLCR-calibrated models.** Improved results are highlighted in **bold**, with positive relative gains shown in red. Mean confidence serves as a reference for analysis alongside other metrics. Since neither higher nor lower mean confidence is inherently better, we do not report its relative improvement.

Math-to-Math	MATH500			AMC			AIME24			AIME25			AIME26		
	ACC ↑	Brier ↓	Conf	ACC ↑	Brier ↓	Conf	ACC ↑	Brier ↓	Conf	ACC ↑	Brier ↓	Conf	ACC ↑	Brier ↓	Conf
RLCR-calibrated Qwen3-4B on DAPO-Math-14K															
Base Model	80.6	0.184	0.984	59.38	0.387	0.984	20.42	0.731	0.951	16.25	0.790	0.979	15.42	0.803	0.974
RLCR Model	89.0	0.090	0.772	79.38	0.134	0.701	29.17	0.228	0.557	24.17	0.238	0.566	23.33	0.257	0.568
w/ **TTCL**	**90.4**	**0.069**	0.913	**88.13**	**0.078**	0.804	**42.08**	**0.160**	0.453	**25.00**	**0.112**	0.360	**24.58**	**0.152**	0.397
# Improv	+1.57%	+23.76%	-	+11.02%	+41.66%	-	+44.29%	+29.60%	-	+3.45%	+52.98%	-	+5.36%	+40.71%	-
RLCR-calibrated Qwen3-8B on DAPO-Math-14K															
Base Model	84.4	0.219	0.987	62.81	0.357	0.983	21.67	0.757	0.978	18.75	0.780	0.981	13.75	0.834	0.981
RLCR Model	90.0	0.052	0.860	80.63	0.092	0.754	35.42	0.106	0.441	28.33	0.069	0.388	27.50	0.127	0.416
w/ **TTCL**	**91.6**	**0.035**	0.881	**89.38**	**0.086**	0.811	**45.42**	**0.101**	0.425	**30.42**	**0.063**	0.390	**28.33**	**0.101**	0.385
# Improv	+1.78%	+31.74%	-	+10.85%	+6.19%	-	+28.24%	+4.38%	-	+7.35%	+8.73%	-	+3.03%	+20.71%	-

Table 13: **Out-of-domain performance of Brier score and mean confidence of TTCL on RLCR-calibrated models.** Improved results are highlighted in **bold**, with positive relative gains shown in red. Mean confidence serves as a reference for analysis alongside other metrics. Since neither higher nor lower mean confidence is inherently better, we do not report its relative improvement.

Math-to-FactQA	SimpleQA			ChineseSimpleQA			NQ			HotpotQA			TriviaQA		
	ACC ↑	Brier ↓	Conf	ACC ↑	Brier ↓	Conf	ACC ↑	Brier ↓	Conf	ACC ↑	Brier ↓	Conf	ACC ↑	Brier ↓	Conf
RLCR-calibrated Qwen3-4B on DAPO-Math-14K															
Base Model	1.40	0.875	0.927	12.13	0.801	0.950	8.63	0.855	0.961	9.56	0.795	0.927	15.63	0.772	0.954
RLCR Model	2.00	0.431	0.643	13.00	0.472	0.730	11.00	0.518	0.762	11.13	0.413	0.671	19.38	0.456	0.745
w/ **TTCL**	**2.70**	**0.134**	0.304	**14.03**	**0.204**	0.447	**12.81**	**0.159**	0.356	**13.50**	**0.176**	0.428	**20.38**	**0.207**	0.494
# Improv	+35.00%	+68.92%	-	+7.95%	+56.77%	-	+16.48%	+69.35%	-	+21.35%	+57.56%	-	+5.16%	+54.55%	-
RLCR-calibrated Qwen3-8B on DAPO-Math-14K															
Base Model	2.20	0.886	0.928	15.77	0.815	0.978	9.63	0.876	0.983	12.75	0.810	0.950	24.13	0.743	0.989
RLCR Model	2.80	0.656	0.796	18.17	0.571	0.824	12.56	0.666	0.862	14.13	0.553	0.777	25.69	0.559	0.862
w/ **TTCL**	**3.90**	**0.136**	0.309	**20.83**	**0.201**	0.434	**15.00**	**0.183**	0.440	**18.38**	**0.219**	0.476	**29.31**	**0.344**	0.720
# Improv	+39.29%	+79.28%	-	+14.68%	+64.75%	-	+19.40%	+72.59%	-	+30.09%	+60.46%	-	+14.11%	+38.44%	-
FactQA-to-Math	**MATH500**			**AMC**			**AIME24**			**AIME25**			**AIME26**		
	ACC ↑	Brier ↓	Conf	ACC ↑	Brier ↓	Conf	ACC ↑	Brier ↓	Conf	ACC ↑	Brier ↓	Conf	ACC ↑	Brier ↓	Conf
RLCR-calibrated Qwen3-4B on FactQA-10K															
Base Model	80.6	0.184	0.984	59.38	0.387	0.984	20.42	0.731	0.951	16.25	0.790	0.979	15.42	0.803	0.974
RLCR Model	79.8	0.126	0.726	64.38	0.182	0.661	23.75	0.268	0.536	18.75	0.235	0.511	14.58	0.263	0.485
w/ **TTCL**	**85.2**	**0.075**	0.827	**80.00**	**0.119**	0.749	**25.00**	**0.153**	0.333	**24.17**	**0.187**	0.354	**16.67**	**0.172**	0.371
# Improv	+6.77%	+40.26%	-	+24.27%	+34.58%	-	+5.26%	+42.80%	-	+28.89%	+20.57%	-	+14.29%	+34.59%	-

combinatorics, and number theory. Compared with MATH500, AMC problems are typically shorter but still require non-trivial reasoning and careful derivation to obtain correct answer.

- **AIME24/AIME25/AIME26** (Zhang & Math-AI, 2024; 2025; 2026): These benchmarks consist of problems from the 2024/2025/2026 American Invitational Mathematics Examination, respectively. These problems are substantially more challenging than standard school-level mathematics questions and typically require several intermediate reasoning steps, making the benchmark useful for evaluating both reasoning accuracy and confidence on difficult mathematical problems.

For factual question answering, we include benchmarks below:

- **SimpleQA** (Wei et al., 2024; Haas et al., 2025): SimpleQA is a challenging factual question answering benchmark proposed by OpenAI, consisting of short fact-seeking questions. In this work, we use the 1,000-question SimpleQA-Verified version introduced by Haas et al. (2025).
- **ChineseSimpleQA** (He et al., 2025): ChineseSimpleQA extends factual question answering evaluation to Chinese and covers a diverse range of knowledge domains. We use it to evaluate calibration and reasoning performance across different language settings.
- **NQ** (Kwiatkowski et al., 2019): NQ is a large-scale open-domain question answering benchmark constructed from real user queries with Wikipedia-based answers. Its questions cover diverse factual topics and formats. We randomly sample 200 examples from the test set for evaluation.
- **HotpotQA** (Yang et al., 2018): HotpotQA is a multi-hop question answering benchmark that requires combining information from multiple supporting facts or documents. We randomly sample 200 examples from the validation set for evaluation.
- **TriviaQA** (Joshi et al., 2017): TriviaQA is a large-scale factual question answering benchmark constructed from trivia questions paired with independently collected evidence documents. We randomly sample 200 examples from the test set as our evaluation set.

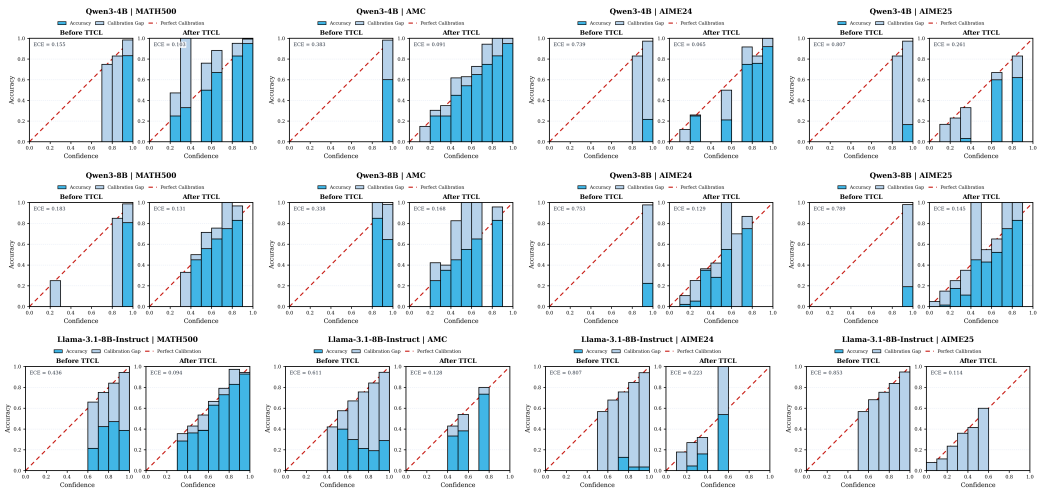


Figure 7: **Visualization of calibration effects before and after TTCL** on base models for mathematical benchmarks. TTCL effectively calibrates models to the target benchmarks, substantially narrowing the gap between confidence and empirical accuracy. Blue bars denote empirical accuracy within each confidence bin, while light blue bars indicate the deviation from perfect calibration. For Llama-3.1-8B-Instruct on AIME25, no accuracy bars is shown because its empirical accuracy is 0.

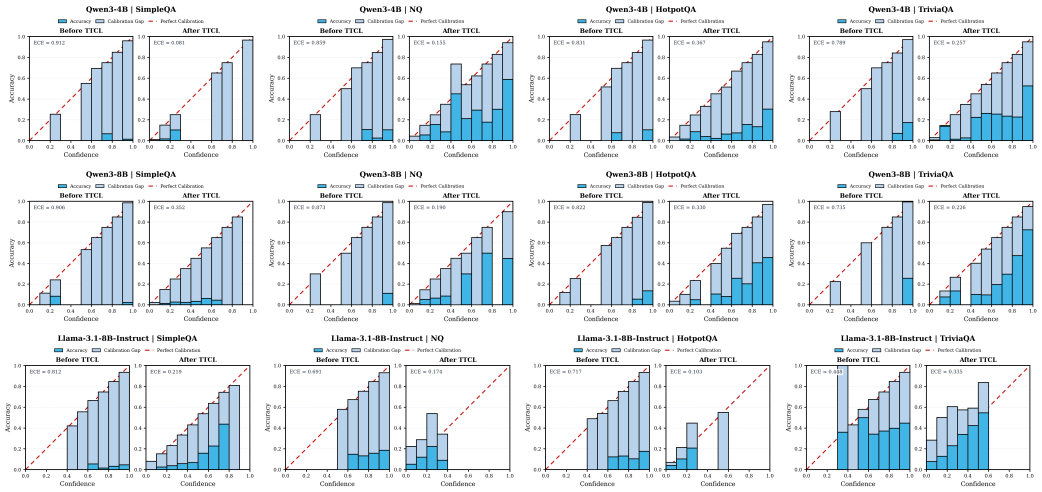


Figure 8: **Visualization of calibration effects before and after TTCL** on base models for factual QA benchmarks. TTCL effectively calibrates models to the target benchmarks, substantially narrowing the gap between confidence and empirical accuracy. Blue bars denote empirical accuracy within each confidence bin, while light blue bars indicate the deviation from perfect calibration.

Additionally, we train RLCR on two labeled datasets, DAPO-Math-14K and FactQA-10K, to obtain calibrated models. The two datasets are introduced as follows:

- **DAPO-Math-14K** (Yu et al., 2025): We construct DAPO-Math-14K using the 14.1K English samples from the DAPO-Math-17k-Processed dataset, a widely used training set for mathematical reasoning containing diverse and high-quality examples with annotated ground-truth answers.
- **FactQA-10K**: We construct FactQA-10K by randomly sampling 5K training examples from HotpotQA (Yang et al., 2018) and 5K training examples from TriviaQA (Joshi et al., 2017), resulting in a total of 10K factual question answering samples.

The benchmark statistics are summarized in Table 9. Additionally, we adopt several benchmarks from diverse domains, including ARC (Clark et al., 2018), GPQA-Diamond (Rein et al., 2023), IFEval (Zhou et al., 2023), MMLU-Pro (Wang et al., 2024), and TruthfulQA (Lin et al., 2022), to

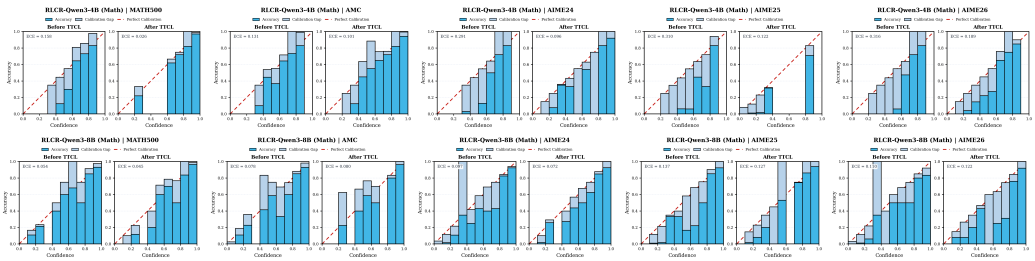


Figure 9: **Visualization of in-domain calibration effects before and after TTCL** on math-calibrated models for mathematical benchmarks. TTCL calibrates models to the target benchmarks, further narrowing the gap between confidence and empirical accuracy. Blue bars denote empirical accuracy within each confidence bin, while light blue bars indicate the deviation from perfect calibration.

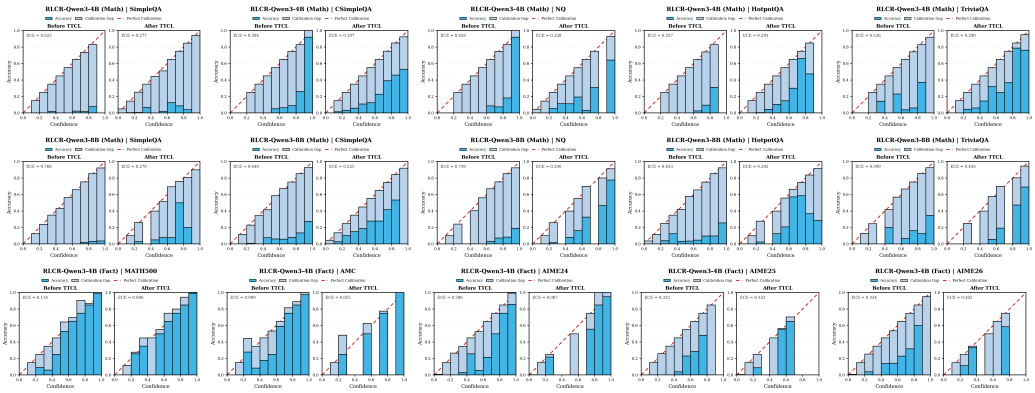


Figure 10: **Visualization of out-of-domain calibration effects before and after TTCL** on math-calibrated models for factual benchmarks. The calibration learned on the source domain transfers poorly to the target domain in both Math-to-FactQA and FactQA-to-Math settings, whereas TTCL effectively adapts the models to the target benchmarks and further narrows the gap between confidence and empirical accuracy. Blue bars denote empirical accuracy within each confidence bin, while light blue bars indicate the deviation from perfect calibration.

examine whether TTCL affects the model's general capabilities beyond the target tasks. These benchmarks cover scientific reasoning, instruction following, multi-domain knowledge and truthfulness. We evaluate the general capabilities of the models on these benchmarks using LightEval (Habib et al., 2023), which is a widely used evaluation platform that supports a broad range of LLM benchmarks.

E.4 More Details of Metrics

In this work, we evaluate both reasoning correctness and confidence calibration mainly using three metrics: Accuracy (ACC), Expected Calibration Error (ECE), and Area Under the Receiver Operating Characteristic Curve (AUROC). Their formulations and details are introduced as follows:

- **ACC:** Accuracy measures the proportion of correctly answered examples and is used to evaluate the model's reasoning performance. Given N evaluation examples, it is computed as

$$\mathrm{ACC} = \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}\left[\mathrm{Ans}(y_i) \equiv a_i^\star\right], \tag{58}$$

  where $\mathbb{I}\left[\mathrm{Ans}(y_i) \equiv a_i^\star\right]$ indicates whether the answer of the i-th response is correct, and $a_i^\star$ is the ground-truth answer. A higher ACC indicates better correctness of the reasoning capability.

- **ECE:** ECE measures the discrepancy between the model's predicted confidence and its empirical correctness. The formulation of ECE is introduced in Eq. (3). A lower ECE indicates better alignment between confidence estimates and actual correctness. Following previous work (Damani et al., 2026; Ma et al., 2026), we set the number of bins to 10 when computing ECE.

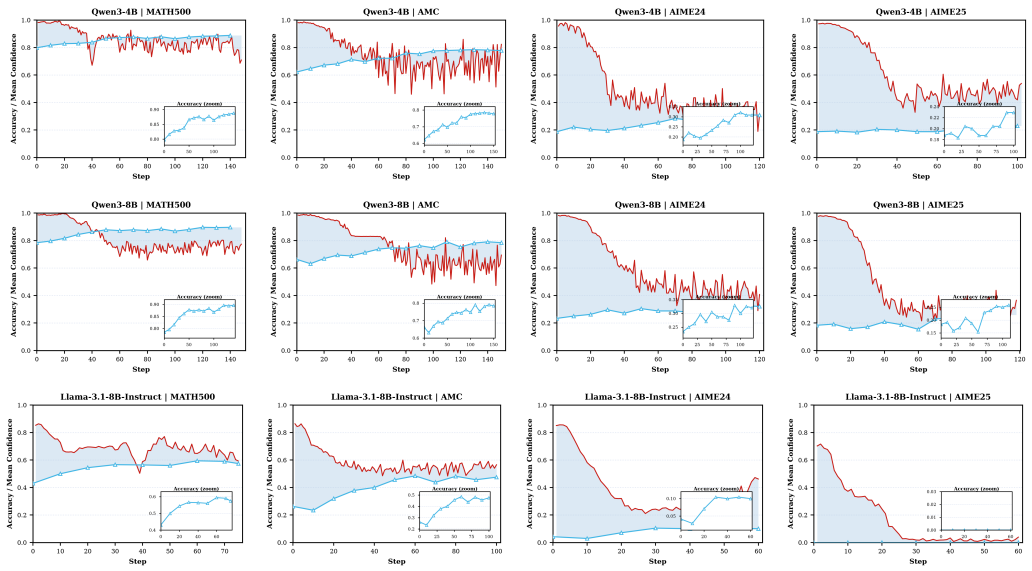


Figure 11: **More training dynamics of TTCL on base models.** The red curves show mean confidence and the cyan-blue curves show empirical accuracy during TTCL training. The shaded area indicates the confidence-accuracy gap, and the inset zooms in on the accuracy trajectory. TTCL progressively aligns confidence with empirical accuracy while improving task accuracy.

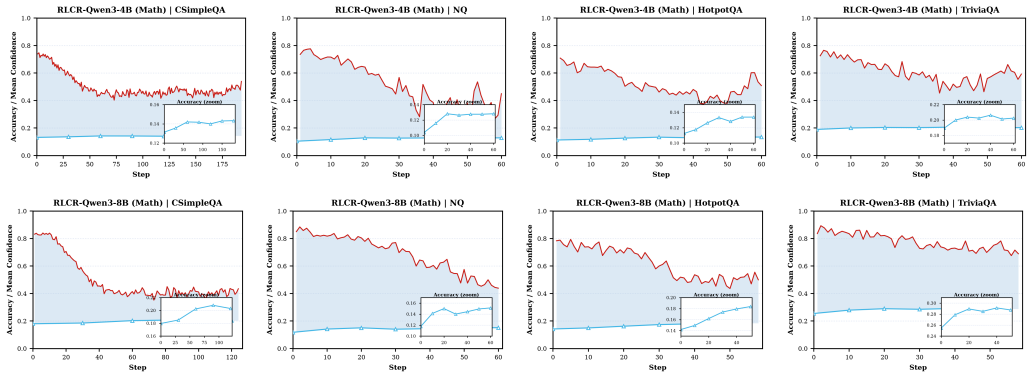


Figure 12: **More training dynamics of TTCL on calibrated models.** The red curves show mean confidence and the cyan-blue curves show empirical accuracy during TTCL training. The shaded area indicates the confidence-accuracy gap, and the inset zooms in on the accuracy trajectory. TTCL progressively aligns confidence with empirical accuracy while improving task accuracy.

- **AUROC:** AUROC measures how well the estimated confidence distinguishes correct responses from incorrect ones. By varying the confidence threshold, we obtain the true-positive rate (TPR) and false-positive rate (FPR), and compute AUROC as follows:

$$\mathrm{AUROC} = \int_0^1 \mathrm{TPR}(u)\, d\mathrm{FPR}(u). \tag{59}$$

  A higher AUROC indicates that correct responses tend to receive higher confidence than incorrect ones. Unlike ECE, which evaluates the absolute alignment between predicted confidence and empirical correctness, AUROC mainly reflects the ranking quality of confidence scores.

For AMC/AIME24/AIME25/AIME26 and NQ/HotpotQA/TriviaQA, due to their relatively small dataset sizes, we sample eight responses per question to reduce the variance of the computed metrics. For MATH500, SimpleQA and ChineseSimpleQA, we sample one response per question.

Table 14: **More results of TTCL-trained models on multiple other domains.** These results examine whether TTCL improves calibration at the cost of model general capabilities on other domains.

TTCL Target	ARC		GPQA-D	IFEval	MMLU-Pro	TruthfulQA		TTCL Target	ARC		GPQA-D	IFEval	MMLU-Pro	TruthfulQA	
	Challenge	Easy				MC1	MC2		Challenge	Easy				MC1	MC2
Qwen3-8B															
Base Model	55.20	75.34	50.00	84.62	63.70	32.80	51.60	**Base Model**	55.20	75.34	50.00	84.62	63.70	32.80	51.60
MATH500	55.29	74.83	51.52	85.47	64.82	33.29	51.65	**SimpleQA**	54.52	74.83	45.96	84.65	63.86	33.17	51.20
AMC	55.29	74.83	45.96	85.83	63.87	33.05	51.35	**NQ**	55.12	74.62	51.52	84.77	63.12	33.05	51.54
AIME24	54.52	74.07	48.99	86.20	62.66	32.68	50.95	**HotpotQA**	55.46	74.79	45.45	86.11	63.87	33.29	51.03
AIME25	54.52	74.12	51.01	84.98	63.16	32.80	50.82	**TriviaQA**	54.78	74.75	48.99	84.56	63.74	32.56	50.70
Qwen3-4B															
Base Model	48.63	65.28	46.97	81.39	56.50	35.74	51.49	**Base Model**	48.63	65.28	46.97	81.39	56.50	35.74	51.49
MATH500	48.98	65.11	46.46	84.48	57.46	35.99	51.66	**SimpleQA**	48.63	65.95	40.91	82.18	57.05	36.23	52.01
AMC	48.55	65.45	45.45	83.29	57.43	36.72	51.94	**NQ**	48.89	64.39	44.95	84.57	57.00	36.23	51.64
AIME24	47.70	64.81	43.94	83.57	56.99	36.23	52.07	**HotpotQA**	48.63	64.56	43.94	82.20	56.86	36.47	51.30
AIME25	48.29	64.81	46.46	83.52	56.39	36.60	51.99	**TriviaQA**	47.95	64.60	42.42	84.92	56.79	35.25	51.70
Llama-3.1-8B-Instruct															
Base Model	59.13	81.23	34.85	77.26	37.73	38.07	52.77	**Base Model**	59.13	81.23	34.85	77.26	37.73	38.07	52.77
MATH500	57.59	79.46	33.33	75.16	37.55	37.58	51.58	**AMC**	55.63	78.87	32.32	75.63	37.73	35.01	49.23
AIME24	58.19	80.01	34.85	77.58	34.92	35.13	49.76	**AIME25**	58.53	80.05	32.83	77.53	36.40	35.74	50.13
RLCR-calibrated Qwen3-8B on DAPO-Math-14K															
Base Model	55.20	75.34	50.00	84.62	63.70	32.80	51.60	**Base Model**	55.20	75.34	50.00	84.62	63.70	32.80	51.60
RLCR Model	55.03	74.62	52.53	84.63	63.85	32.44	51.48	**RLCR Model**	55.03	74.62	52.53	84.63	63.85	32.44	51.48
MATH500	54.95	74.33	52.02	84.01	64.44	32.93	51.63	**SimpleQA**	55.03	74.03	52.53	84.28	64.07	32.93	50.49
AMC	54.61	74.41	57.07	84.58	64.50	32.80	51.61	**ChineseSimpleQA**	54.95	74.16	55.05	84.10	64.54	32.56	50.83
AIME24	54.44	73.53	52.53	85.28	64.15	32.68	50.51	**NQ**	54.69	74.20	52.53	84.74	64.06	32.56	51.12
AIME25	55.03	74.20	50.00	83.46	64.39	33.17	51.32	**HotpotQA**	54.86	74.33	52.53	85.38	64.40	32.93	51.22
AIME26	54.61	74.16	52.53	84.08	64.35	32.93	51.27	**TriviaQA**	54.86	74.49	53.03	84.07	63.92	32.56	50.71
RLCR-calibrated Qwen3-4B on DAPO-Math-14K															
Base Model	48.63	65.28	46.97	81.39	56.50	35.74	51.49	**Base Model**	48.63	65.28	46.97	81.39	56.50	35.74	51.49
RLCR Model	48.63	65.45	50.51	82.70	58.75	36.72	51.84	**RLCR Model**	48.63	65.45	50.51	82.70	58.75	36.72	51.84
MATH500	48.04	64.52	47.98	82.68	59.52	35.86	51.71	**SimpleQA**	49.06	65.78	48.99	82.21	59.56	35.74	51.75
AMC	48.63	64.77	51.01	81.92	58.21	36.23	52.11	**ChineseSimpleQA**	48.46	64.56	47.47	83.08	58.30	36.60	51.79
AIME24	47.95	64.69	48.99	83.50	59.14	36.47	51.98	**NQ**	48.29	64.81	48.48	83.61	59.86	36.35	51.94
AIME25	48.12	64.98	49.49	84.36	58.64	36.35	52.00	**HotpotQA**	48.63	65.24	46.97	83.06	58.89	36.35	52.26
AIME26	48.04	65.15	48.99	83.61	59.00	36.47	52.03	**TriviaQA**	48.38	65.32	53.03	82.91	58.88	36.23	51.46
RLCR-calibrated Qwen3-4B on FactQA-10K															
Base Model	48.63	65.28	46.97	81.39	56.50	35.74	51.49	**RLCR Model**	49.15	65.53	43.43	83.20	57.10	35.86	51.27
RLCR Model	49.15	65.53	43.43	83.20	57.10	35.86	51.27	**AIME24**	48.55	65.32	44.44	83.08	56.85	36.35	51.70
MATH500	48.63	65.36	43.94	82.88	56.92	35.99	51.27	**AIME25**	47.61	64.27	45.96	83.16	56.63	36.23	51.48
AMC	48.72	65.49	43.94	84.10	57.58	36.11	51.90	**AIME26**	48.29	65.03	44.95	83.52	56.88	36.35	51.72

F Additional Experimental Results

F.1 More Results of Calibration Performance

As a supplement, Table 10, Table 11, Table 12 and Table 13 additionally introduce the metrics of Brier score and mean confidence. The Brier score measures the squared discrepancy between predicted confidence and actual correctness, with a lower score indicating better confidence calibration. The mean confidence, when considered together with accuracy, provides a complementary view of the model's overall confidence tendency toward over-confidence or under-confidence.

Existing base models exhibit substantial over-confidence across both mathematical reasoning and factual QA benchmarks. As shown in Table 10 and Table 11, it can be observed that the mean confidence of the base models is often considerably higher than their empirical accuracy. This mismatch is especially evident on several benchmarks where the models maintain high confidence despite relatively low correctness, resulting in large Brier scores. These results further motivate the need for calibration learning of the models to provide reliable confidence estimates.

TTCL consistently lowers the Brier score and makes mean confidence more closely aligned with empirical accuracy. Across all four tables, it can be observed that TTCL consistently reduces the Brier score and narrows the gap between accuracy and mean confidence, providing additional evidence that TTCL indeed performs effective calibration learning for the models. For results of the base models in Table 10 and Table 11, TTCL effectively corrects the mismatch between confidence and empirical accuracy across both mathematical reasoning and factual QA benchmarks, while simultaneously improving answer accuracy. For results of the RLCR-calibrated models in Table 12 and Table 13, TTCL further improves calibration on top of the already calibrated models. In particular, under out-of-domain, where the confidence learned from the source domain may transfer poorly on the target domain, TTCL effectively adapts the confidence estimates to the target distribution.

TTCL does not simply decrease confidence; instead, it adjusts confidence toward the model's empirical accuracy. For example, for the RLCR-calibrated Qwen3-4B on MATH500 in Table 12, the mean confidence increases from 0.772 to 0.913 after TTCL, closely matching the corresponding accuracy of 90.4%. This result indicates that the calibration improvement achieved by TTCL is not due to indiscriminately suppressing confidence, but rather to better aligning confidence with empirical

Table 15: **Detailed IFEval results of TTCL-trained models.** IFEval reports four metrics: prompt-level strict accuracy, prompt-level loose accuracy, instruction-level strict accuracy, and instruction-level loose accuracy. In previous tables, we report the average of these four metrics. Here, we provide the detailed results as a supplement to Table 6 and Table 14.

TTCL Target	Avg.	Prompt Strict	Inst. Strict	Prompt Loose	Inst. Loose	TTCL Target	Avg.	Prompt Strict	Inst. Strict	Prompt Loose	Inst. Loose
						Qwen3-8B					
Base Model	84.62	80.04	86.45	83.36	88.61	**Base Model**	84.62	80.04	86.45	83.36	88.61
MATH500	85.47	81.15	87.05	84.47	89.21	**SimpleQA**	84.65	79.85	86.33	83.55	88.85
AMC	85.83	81.33	87.29	85.03	89.69	**NQ**	84.77	80.41	83.36	86.69	88.61
AIME24	86.20	82.07	87.89	85.03	89.81	**HotpotQA**	86.11	82.07	85.03	87.65	89.69
AIME25	84.98	80.59	86.57	83.92	88.85	**TriviaQA**	84.56	80.22	83.18	86.33	88.49
						Qwen3-4B					
Base Model	81.39	75.97	83.45	80.04	86.09	**Base Model**	81.39	75.97	83.45	80.04	86.09
MATH500	84.48	80.04	85.97	83.55	88.37	**SimpleQA**	82.18	76.89	84.05	80.96	86.81
AMC	83.29	78.37	84.77	82.26	87.77	**NQ**	84.58	80.41	83.55	85.97	88.37
AIME24	83.57	78.74	85.13	82.62	87.77	**HotpotQA**	82.20	77.26	81.15	83.81	86.57
AIME25	83.52	78.19	84.77	82.99	88.13	**TriviaQA**	84.93	80.59	84.29	86.09	88.73
						Llama-3.1-8B-Instruct					
Base Model	77.26	70.98	79.74	75.23	83.09	**Base Model**	77.26	70.98	79.74	75.23	83.09
MATH500	75.16	68.21	77.10	73.94	81.41	**AMC**	75.63	69.50	78.18	73.57	81.29
AIME24	77.58	71.35	79.86	75.79	83.33	**AIME25**	77.53	70.98	80.10	75.60	83.45
						RLCR-calibrated Qwen3-8B on DAPO-Math-14K					
Base Model	84.62	80.04	86.45	83.36	88.61	**Base Model**	84.62	80.04	86.45	83.36	88.61
RLCR Model	84.63	80.22	86.69	82.99	88.61	**RLCR Model**	84.63	80.22	86.69	82.99	88.61
MATH500	84.01	79.67	86.09	82.26	88.01	**SimpleQA**	84.28	79.85	86.21	82.81	88.25
AMC	84.58	79.67	86.45	83.36	88.85	**ChineseSimpleQA**	84.10	79.30	85.85	82.99	88.25
AIME24	85.28	80.96	87.17	83.92	89.09	**NQ**	84.74	80.59	86.45	83.55	88.37
AIME25	83.46	78.74	85.73	81.70	87.65	**HotpotQA**	85.38	81.15	87.17	84.10	89.09
AIME26	84.08	79.67	86.09	82.44	88.13	**TriviaQA**	84.07	79.67	85.97	82.62	88.01
						RLCR-calibrated Qwen3-4B on DAPO-Math-14K					
Base Model	81.39	75.97	83.45	80.04	86.09	**Base Model**	81.39	75.97	83.45	80.04	86.09
RLCR Model	82.70	77.08	84.29	81.89	87.53	**RLCR Model**	82.70	77.08	84.29	81.89	87.53
MATH500	82.68	77.63	84.29	81.52	87.29	**SimpleQA**	82.21	77.08	84.17	80.78	86.81
AMC	81.92	76.52	83.45	81.15	86.57	**ChineseSimpleQA**	83.08	78.19	84.77	82.07	87.29
AIME24	83.50	78.37	85.01	82.62	88.01	**NQ**	83.61	78.74	85.13	82.81	87.77
AIME25	84.36	79.67	85.85	83.55	88.37	**HotpotQA**	83.06	78.19	84.89	81.89	87.29
AIME26	83.61	78.74	85.13	82.81	87.77	**TriviaQA**	82.91	77.26	83.93	82.81	87.65
						RLCR-calibrated Qwen3-4B on FactQA-14K					
Base Model	81.39	75.97	83.45	80.04	86.09	**RLCR Model**	83.20	78.56	84.77	82.07	87.41
RLCR Model	83.20	78.56	84.77	82.07	87.41	**AIME24**	83.08	78.00	84.77	81.89	87.65
MATH500	82.88	77.82	84.53	81.89	87.29	**AIME25**	83.16	78.19	84.65	82.26	87.53
AMC	84.10	79.48	85.49	83.18	88.25	**AIME26**	83.52	78.74	85.13	82.44	87.77

correctness. This observation provides empirical evidence for our theoretical analysis that the TTCL objective serves as a surrogate for the ideal calibration learning objective.

These results further support our analysis in Section 5.2 that TTCL can improve both accuracy and calibration on target tasks at test time, aligning confidence estimates with empirical correctness.

F.2 More Visualizations of TTCL Calibration Effects

As a supplement to the results in Fig. 6, to provide an intuitive view of the calibration effects of TTCL, we present additional visualizations under different settings in Fig. 7, Fig. 8, Fig. 9 and Fig. 10. Specifically, we visualize the calibration performance before and after TTCL for: (1) base models on mathematical reasoning benchmarks (Fig. 7), (2) base models on factual QA benchmarks (Fig. 8), (3) math-calibrated models on mathematical reasoning benchmarks (Fig. 9), and (4) cross-domain adaptation, including math-calibrated models (DAPO-Math-14K) on factual QA benchmarks and fact-calibrated models (FactQA-10K) on mathematical reasoning benchmarks (Fig. 10).

Across these settings, TTCL consistently improves the alignment between confidence and empirical accuracy. For base models, TTCL effectively corrects their pronounced over-confidence, bringing confidence into closer alignment with empirical accuracy. For math-calibrated models evaluated in-domain, TTCL further refines calibration on top of the existing calibrated initialization. More importantly, under cross-domain transfer, calibration learned on the source domain transfers poorly to the target domain, whereas TTCL effectively adapts confidence estimates to the target benchmarks. Overall, these visualizations provide an intuitive view of the consistent calibration improvements achieved by TTCL across different model initializations and domain settings.

F.3 More Training Dynamics of TTCL

As a supplement to Fig. 5, we provide additional training dynamics to illustrate how TTCL adjusts model confidence and accuracy throughout test-time learning. Figure 11 shows the dynamics on base models, and Figure 12 presents the corresponding dynamics on math-calibrated models under

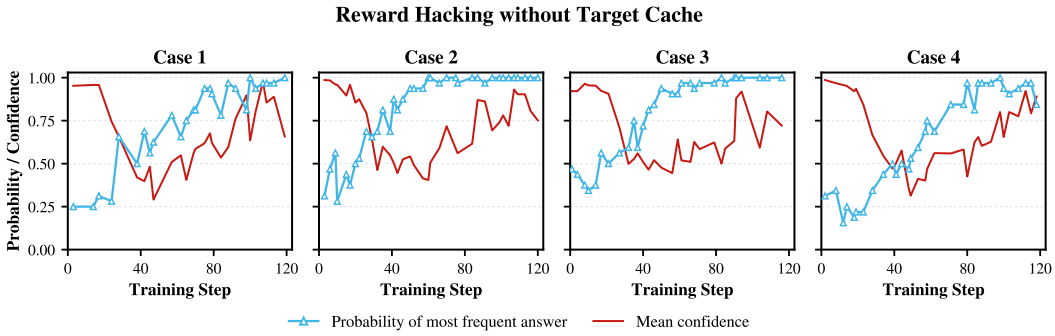


Figure 13: **Reward hacking without the target cache.** Four AIME24 examples in which the model's rollouts increasingly concentrate on an incorrect answer while mean confidence rises toward 1.0. The blue lines show the frequency-based probability of the most frequent answer, and the red lines show mean confidence. When supervision signals are derived from the the rollout group of current policy, increasing agreement on a wrong answer raises its confidence target, leading to reward hacking.

factual QA adaptation. It can be observed that, across different models and target benchmarks, TTCL consistently drives mean confidence toward empirical accuracy while improving task accuracy. For base models, the initially pronounced over-confidence is progressively corrected during training, leading to a substantially reduced confidence-accuracy gap. For calibrated models, TTCL further adapts the existing confidence estimates to the target-domain performance, showing that calibration learned from the source setting can continue to be refined at test time. Overall, these dynamics provide a process-level view of how TTCL jointly improves accuracy and calibration during adaptation, further supporting our theoretical analysis that the TTCL objective serves as a surrogate for the ideal calibration learning objective by aligning confidence estimates with empirical correctness.

F.4 Reward Hacking without Target Cache

To examine the failure mode that motivates the target cache, we train Qwen3-4B on AIME24 using supervision computed directly from the current rollout group. Fig. 13 shows four questions for which the rollouts increasingly concentrate on an incorrect answer. As more rollouts produce that answer, its frequency-based probability rises toward one. Because this probability also serves as the confidence target, the supervision encourages the model's confidence to rise toward 1.0 despite the answer being wrong. This is a form of reward hacking: the model can obtain a higher self-generated reward by consistently producing an incorrect answer with high confidence, without improving correctness or calibration. The target cache is motivated by this failure mode; by fixing targets within each epoch, it weakens the immediate feedback between the current policy and its self-supervision signals.

F.5 More Results of General Capabilities

As a supplement to Table 6 in Section 5.3, we provide more comprehensive results in Table 14. Specifically, we further report results for more TTCL-trained models on benchmarks covering multiple aspects of general capabilities, including ARC (Clark et al., 2018), GPQA-Diamond (Rein et al., 2023), IFEval (Zhou et al., 2023), MMLU-Pro (Wang et al., 2024), and TruthfulQA (Lin et al., 2022), which evaluate capabilities across scientific reasoning, instruction following, multi-domain knowledge, and truthfulness. For IFEval, the evaluation includes four metrics: prompt-level strict, prompt-level loose, instruction-level strict, and instruction-level loose accuracy; we report their average values, with the detailed results provided in Appendix F.6. Across these benchmarks, TTCL-trained models generally maintain performance comparable to their corresponding models before TTCL adaptation, with only minor fluctuations across individual tasks. These results indicate that the calibration improvements brought by TTCL do not come at the cost of degrading the model's general capabilities on other domains. Instead, TTCL can improve accuracy and confidence calibration on the target tasks while largely preserving the model's general capabilities.

Table 16: **Calibration results of existing label-free RL methods.** We evaluate several label-free RL methods using the released checkpoints from Zhang et al. (2026), and report accuracy, ECE, and mean confidence values. Mean confidence values that are substantially higher than the corresponding empirical accuracy, indicating over-confidence of calibration, are highlighted in red.

Label-free RL Methods	MATH500			AMC			AIME24			AIME25			AIME26		
	ACC ↑	ECE ↓	Conf	ACC ↑	ECE ↓	Conf	ACC ↑	ECE ↓	Conf	ACC ↑	ECE ↓	Conf	ACC ↑	ECE ↓	Conf
Qwen3-4B-Base															
Base Model	58.8	0.347	0.935	38.44	0.541	0.926	10.00	0.826	0.926	4.17	0.876	0.918	6.25	0.861	0.924
Self-Certainty (Zhao et al., 2026)	61.6	0.342	0.958	41.25	0.540	0.953	7.50	0.877	0.952	5.00	0.894	0.944	5.00	0.900	0.950
Entropy (Prabhudesai et al., 2025)	77.2	0.206	0.978	51.88	0.462	0.981	14.58	0.833	0.978	8.75	0.890	0.978	9.58	0.885	0.980
Majority-voting (Zuo et al., 2025)	72.6	0.230	0.956	43.13	0.509	0.940	9.58	0.831	0.927	4.58	0.876	0.921	5.83	0.864	0.923
Co-rewarding (Zhang et al., 2026)	75.6	0.214	0.970	52.81	0.431	0.959	14.58	0.801	0.947	9.17	0.852	0.944	6.67	0.881	0.947
Qwen3-8B-Base															
Base Model	60.6	0.305	0.911	42.50	0.487	0.912	13.75	0.773	0.911	13.33	0.778	0.912	6.25	0.846	0.908
Self-Certainty (Zhao et al., 2026)	83.2	0.132	0.964	59.69	0.363	0.960	20.00	0.750	0.950	16.25	0.787	0.949	13.33	0.820	0.954
Entropy (Prabhudesai et al., 2025)	79.0	0.177	0.967	57.50	0.388	0.963	14.17	0.818	0.960	14.17	0.813	0.955	9.17	0.866	0.957
Majority-voting (Zuo et al., 2025)	78.2	0.180	0.962	57.19	0.380	0.952	12.08	0.822	0.943	7.50	0.862	0.937	6.25	0.875	0.937
Co-rewarding (Zhang et al., 2026)	81.0	0.157	0.967	60.63	0.350	0.956	15.00	0.797	0.947	11.25	0.834	0.946	11.67	0.829	0.945
Llama-3.2-3B-Instruct															
Base Model	35.2	0.514	0.866	24.06	0.615	0.855	6.67	0.766	0.832	0.00	0.828	0.828	0.83	0.825	0.834
Self-Certainty (Zhao et al., 2026)	47.0	0.428	0.898	25.00	0.637	0.887	6.67	0.807	0.874	0.42	0.865	0.869	0.42	0.863	0.867
Entropy (Prabhudesai et al., 2025)	31.0	0.589	0.899	15.63	0.730	0.887	5.00	0.815	0.865	0.42	0.866	0.870	0.83	0.850	0.858
Majority-voting (Zuo et al., 2025)	42.0	0.470	0.890	25.63	0.619	0.875	7.50	0.781	0.856	0.42	0.846	0.850	0.83	0.838	0.847
Co-rewarding (Zhang et al., 2026)	46.4	0.426	0.890	28.44	0.561	0.846	7.08	0.743	0.814	0.00	0.813	0.813	1.25	0.801	0.814

F.6 Detailed Results of IFEval

In the previous results, we report the average performance over the four IFEval metrics: prompt-level strict accuracy, prompt-level loose accuracy, instruction-level strict accuracy, and instruction-level loose accuracy. As a supplement, Table 15 further reports the detailed results for each individual metric. It can be observed that TTCL effectively preserves the model's instruction-following capability. Across different target tasks and model settings, the IFEval performance after TTCL remains generally comparable to that before TTCL training, with slight improvements observed in several cases, such as on Qwen3-4B. These results further indicate that TTCL can improve accuracy and calibration on target tasks without compromising the model's general instruction-following capability.

F.7 Calibration Results of Existing Label-free RL Methods

To examine whether existing label-free RL methods can improve model calibration while enhancing reasoning accuracy, we evaluate several representative approaches, including Self-certainty Maximization (Zhao et al., 2026), Entropy Minimization (Prabhudesai et al., 2025), Majority-voting for pseudo-label construction (Zuo et al., 2025), and Co-rewarding (Zhang et al., 2026). These methods are primarily designed to improve reasoning performance without ground-truth labels, but lack an explicit calibration objective. Thanks to the Co-rewarding work, which releases checkpoints trained on DAPO-Math-14K with these different label-free RL methods, we directly evaluate these checkpoints for calibration performance. Since most of them are trained from pretrained models and have limited ability to follow verbalized-confidence instructions, we use the mean token probability of the generated response as the confidence estimate (Malinin & Gales, 2020; Orgad et al., 2025).

As shown in Table 16, although these label-free RL methods improve accuracy to some extent compared with base models, their calibration remains relatively poor. We observe that the ECE values are generally high, while the mean confidence consistently exceeds 0.8 and is typically above 0.9 for the Qwen models, indicating clear over-confidence. These results suggest that accuracy-oriented label-free RL does not naturally lead to well-calibrated confidence estimates when an explicit calibration objective is absent. These observations are consistent with our discussion in Appendix B Related Work and further motivate our exploration of label-free test-time calibration learning framework.

F.8 Case Study

To intuitively illustrate the calibration behavior learned by TTCL, we present several cases after TTCL adaptation. In these case studies, blue boxes denote the questions, red boxes denote outputs that are not well calibrated (e.g., over-confidence, incorrect answer with high confidence), and green boxes denote the well-calibrated outputs after TTCL. These cases show that the TTCL-trained models can effectively assign high confidence to correct answers while expressing relatively low confidence on incorrect answers. When calibration learned on a source domain transfers poorly

to newly encountered target tasks, TTCL can further adapt the model's confidence to the target distribution and improve its calibration on these tasks. This provides a concrete view of how TTCL aligns verbalized confidence with prediction correctness on target tasks.

Question from AMC:

For how many integers n does the expression

$$\sqrt{\frac{\log(n^2) - (\log n)^2}{\log n - 3}}$$

represent a real number, where log denotes the base 10 logarithm?
Ground-truth answer: $\boxed{901}$

Over-confidence of Qwen3-4B (Incorrect Answer with High Confidence)

We are given the expression:

$$\sqrt{\frac{\log(n^2) - (\log n)^2}{\log n - 3}}$$

We are to find how many **integers** n make this expression a **real number**, where $\log$ denotes the **base 10 logarithm**.

Step 1: Simplify the expression
Let's denote $x = \log n$. Since $\log$ is base 10, the expression becomes:

$$\sqrt{\frac{\log(n^2) - (\log n)^2}{\log n - 3}} = \sqrt{\frac{2\log n - (\log n)^2}{\log n - 3}} = \sqrt{\frac{2x - x^2}{x - 3}}$$

We want this to be **real**, so the **expression inside the square root must be non-negative**:

$$\frac{2x - x^2}{x - 3} \geq 0$$

Also, the **denominator cannot be zero**, so:

$$x \neq 3 \Rightarrow \log n \neq 3 \Rightarrow n \neq 10^3 = 1000$$

So, we must have:

$$\frac{2x - x^2}{x - 3} \geq 0, \quad x \neq 3$$

Step 2: Solve the inequality
We analyze the sign of the expression:

$$\frac{2x - x^2}{x - 3} = \frac{-x^2 + 2x}{x - 3} = \frac{-x(x - 2)}{x - 3}$$

Let's find the critical points:

- Numerator: $-x(x - 2) = 0 \Rightarrow x = 0, 2$
- Denominator: $x - 3 = 0 \Rightarrow x = 3$

So the sign chart is defined by intervals:

- $x < 0$
- $0 < x < 2$
- $2 < x < 3$
- $x > 3$

Now test the sign of the expression in each interval:

Interval	**Sign of $-x$**	**Sign of $x-2$**	**Sign of $x-3$**	**Sign of $\frac{-x(x-2)}{x-3}$**
$x<0$	$-$	$-$	$-$	$\frac{(-)(-)}{(-)}=\frac{+}{-}=-$
$0<x<2$	$-$	$-$	$-$	$\frac{(-)(-)}{(-)}=\frac{+}{-}=-$
$2<x<3$	$-$	$+$	$-$	$\frac{(-)(+)}{(-)}=\frac{-}{-}=+$
$x>3$	$-$	$+$	$+$	$\frac{(-)(+)}{(+)}=\frac{-}{+}=-$

So the expression is **non-negative** when:

$$2<x<3$$

But $x=\log n$, so:

$$2<\log n<3 \Rightarrow 10^2<n<10^3 \Rightarrow 100<n<1000$$

Also, n must be an **integer**, and $n\neq 1000$ (since $\log n=3$ is not allowed).

Step 3: Count the integers in the interval
We are to count the integers n such that:

$$100<n<1000$$

So n ranges from 101 to 999 (inclusive). The number of such integers is:

$$999-101+1=899$$

Final Answer:

$$\boxed{899}$$

<conf> Confidence: 0.95

Well-calibrated Qwen3-4B after TTCL on AMC (Incorrect Answer with Low Confidence)

To determine for how many integers n the expression

$$\sqrt{\frac{\log(n^2)-(\log n)^2}{\log n-3}}$$

represents a real number, we need to ensure two conditions:

1. The expression inside the square root must be **non-negative** (i.e., greater than or equal to 0).
2. The denominator $\log n-3$ must not be **zero** (to avoid division by zero).

Let's analyze the expression step by step.

Step 1: Simplify the numerator
We are given:

$$\log(n^2)-(\log n)^2$$

Using the logarithmic identity $\log(n^2)=2\log n$, this becomes:

$$2\log n-(\log n)^2$$

Let $x=\log n$. Then the numerator becomes:

$$2x-x^2=-x^2+2x$$

So the entire expression becomes:

$$\sqrt{\frac{-x^2+2x}{x-3}}$$

Where $x = \log n$.

Step 2: Analyze the expression inside the square root
We need:

$$\frac{-x^2+2x}{x-3} \geq 0$$

Also, we must have $x \neq 3$ (to avoid division by zero).
Let's analyze the sign of the expression:

$$\frac{-x^2+2x}{x-3} = \frac{-x(x-2)}{x-3}$$

We now analyze the sign of this expression.

Step 3: Find the critical points
The numerator is zero when $x = 0$ or $x = 2$.
The denominator is zero when $x = 3$.
So the critical points are $x = 0, 2, 3$.
We now test the sign of the expression in the intervals determined by these points:

- $x < 0$
- $0 < x < 2$
- $2 < x < 3$
- $x > 3$

Let's analyze the sign of the expression in each interval.
Interval 1: $x < 0$

- Numerator: $-x(x-2)$: $x < 0$, so $-x > 0$, and $x - 2 < 0$, so overall numerator is positive.
- Denominator: $x - 3 < 0$
- Overall: $\frac{+}{-} = - \rightarrow$ Negative $\rightarrow$ Not allowed.

Interval 2: $0 < x < 2$

- Numerator: $-x(x-2)$: $x > 0$, $x - 2 < 0$, so overall numerator is positive.
- Denominator: $x - 3 < 0$
- Overall: $\frac{+}{-} = - \rightarrow$ Negative $\rightarrow$ Not allowed.

Interval 3: $2 < x < 3$

- Numerator: $-x(x-2)$: $x > 0$, $x - 2 > 0$, so overall numerator is negative.
- Denominator: $x - 3 < 0$
- Overall: $\frac{-}{-} = + \rightarrow$ Positive $\rightarrow$ Allowed.

Interval 4: $x > 3$

- Numerator: $-x(x-2)$: $x > 0$, $x - 2 > 0$, so overall numerator is negative.
- Denominator: $x - 3 > 0$
- Overall: $\frac{-}{+} = - \rightarrow$ Negative $\rightarrow$ Not allowed.

Step 4: Valid interval
The expression is **non-negative** only in the interval $2 < x < 3$.
So we require:

$$2 < \log n < 3$$

Convert to n:

$$10^2 < n < 10^3 \Rightarrow 100 < n < 1000$$

Now, n must be an **integer** in this interval.
The integers from 101 to 999 inclusive are:

$$999 - 101 + 1 = 899$$

But we must also ensure that $\log n \neq 3$, which means $n \neq 1000$.
So the valid integers are from 101 to 999 inclusive.

$$999 - 101 + 1 = 899$$

Thus, the number of integers n for which the expression represents a real number is:

$$\boxed{899}$$

<conf> Confidence: 0.25

Question from AIME24:

Jen enters a lottery by picking 4 distinct numbers from $S = \{1, 2, 3, \cdots, 9, 10\}$. 4 numbers are randomly chosen from S. She wins a prize if at least two of her numbers were 2 of the randomly chosen numbers, and wins the grand prize if all four of her numbers were the randomly chosen numbers. The probability of her winning the grand prize given that she won a prize is $\frac{m}{n}$ where m and n are relatively prime positive integers. Find $m + n$.
Ground-truth answer: $\boxed{116}$

Well-calibrated Qwen3-4B after TTCL on AIME24 (Correct Answer with High Confidence)

To solve this problem, we need to calculate the probability that Jen wins the **grand prize** given that she has won a **prize** (i.e., at least two of her numbers match the randomly chosen numbers).

Step 1: Total number of ways to choose 4 numbers from 10

$$\binom{10}{4} = 210$$

Step 2: Number of ways Jen can win a prize (at least 2 of her numbers match)
This includes:

- Exactly 2 matching numbers
- Exactly 3 matching numbers
- Exactly 4 matching numbers (grand prize)

Let's calculate each case.

Case 1: Exactly 2 matching numbers

- Choose 2 numbers from Jen's 4: $\binom{4}{2}$
- Choose 2 numbers from the remaining 6: $\binom{6}{2}$

$$\binom{4}{2} \cdot \binom{6}{2} = 6 \cdot 15 = 90$$

Case 2: Exactly 3 matching numbers

- Choose 3 numbers from Jen's 4: $\binom{4}{3}$
- Choose 1 number from the remaining 6: $\binom{6}{1}$

$$\binom{4}{3} \cdot \binom{6}{1} = 4 \cdot 6 = 24$$

Case 3: Exactly 4 matching numbers (grand prize)

- Choose all 4 numbers from Jen's 4: $\binom{4}{4} = 1$

Total number of ways to win a prize:

$$90 + 24 + 1 = 115$$

Step 3: Number of ways to win the grand prize (exactly 4 matching numbers)

$$1$$

Step 4: Probability of winning the grand prize given that she won a prize

$$P(\text{grand prize} \mid \text{prize}) = \frac{1}{115}$$

Step 5: Simplify the fraction $\frac{1}{115}$
The fraction is already in simplest form, so $m = 1$, $n = 115$.

Step 6: Find $m + n$

$$1 + 115 = 116$$

Final Answer:

$$\boxed{116}$$

<conf> Confidence: 0.92

Question from factual benchmark NQ:

when is season 2 of jessica jones being released
Ground-truth answer: March 8, 2018

RLCR-Qwen3-8B calibrated on math transfers poorly on FactQA (Incorrect Answer with High Confidence)

Let's break this down step by step:

1. **Identify the show**: *Jessica Jones* is a Marvel superhero TV series that aired on Netflix. It is part of the *Marvel Netflix Universe* (MNLU).
2. **Season 1 release**: *Jessica Jones* Season 1 was released on **November 20, 2015**.
3. **Season 2 release**: Season 2 was released on **March 15, 2017**.
4. **Current status**: The show was canceled by Netflix in 2018, and no further seasons were produced. However, *Jessica Jones* was later picked up by Marvel Studios for a film, *Jessica Jones* (2020), which is not a continuation of the TV series.

Final Answer:
Season 2 of *Jessica Jones* was released on **March 15, 2017**.
<answer>March 15, 2017</answer>
<conf> Confidence: 0.92

TTCL further calibrates RLCR-Qwen3-8B on the target NQ task (Incorrect Answer with Low Confidence)

Jessica Jones is a Marvel Netflix series that originally aired in 2015. Season 2 of *Jessica Jones* was released on **December 1, 2016**.
The series was produced by Marvel Television and released on Netflix, so the release date is specific to the streaming platform.
<answer>December 1, 2016</answer>
<conf> Confidence: 0.1

Question from factual benchmark NQ:

what's the dog's name on tom and jerry
Ground-truth answer: Spike

TTCL further calibrates RLCR-Qwen3-8B on the target NQ task (Correct Answer with High Confidence):

The dog's name on *Tom and Jerry* is **Spike**.
Tom is the cat, and Spike is the dog. They are the main characters in the classic animated series.
<answer>Spike</answer>
<conf> Confidence: 0.95