Cross-Question Reliable Reinforcement Learning

Hector G. Rodriguez **Marcus Rohrbach**
TU Darmstadt & hessian.AI, Germany

Abstract

AI systems operating near the limits of their capabilities can maximize utility while minimizing the risk of error by abstaining or deferring questions they are not confident about to a human or a stronger model. Recently, large language models (LLMs) have been post-trained to estimate confidence in their own answers. To achieve this, methods reinforce confidence scores based on the correctness signal to each question-answer pair *individually*, with rewards that encourage assigning confidence 1 to correct answers and 0 to incorrect answers. However, estimating confidence scores is useful in that it enables selecting only the most confident answers, to maximize the share of solved questions while accommodating the user's risk tolerance. Assigning confidence 1 to correct answers and 0 to incorrect ones is theoretically optimal, but unattainable in practice and detrimental for selective prediction with a non-zero risk tolerance: confidences pushed to 0 and 1 carry no information about how answers compare to one another. We therefore introduce CARE (Cross-question Abstention Reward Estimation), which computes rewards over a set of question-answer pairs: At each of several risk tolerances, the confidences across the set induce an abstention threshold, and each confidence is rewarded for lying on the correct side of it. Since the threshold itself comes from the set, every verbalized confidence is reinforced relative to other questions' confidence, not against its own binary correctness label. On math, CARE increases the coverage at 5% risk (C@5) across MATH-500, GSM8K, and Big-Math-Digits from 0.8% to 42.4%, compared to existing pointwise correctness-matching rewards. When trained on multi-hop Wikipedia QA from HotpotQA, CARE improves C@5 in out-of-distribution general-knowledge, reasoning, and math benchmarks (including CommonsenseQA, GPQA, SimpleQA, and TriviaQA) from 12.3% to 23.6%.

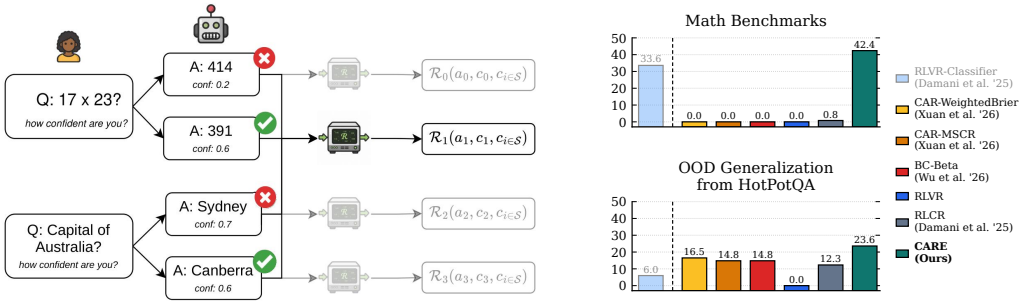


Figure 1: **Cross-Question Abstention Reward Estimation (CARE).** LLMs are trained to verbalize confidence in their answers. Existing pointwise rewards score predicted confidence only against its correctness label, i.e. confidence reward $\mathcal{R}_i(a_i, c_i)$ for answer a_i depends only on that answer's correctness and confidence prediction. **Left:** CARE uses sets of answers and confidences $\mathcal{S}$ across a batch to reinforce coverage at risk, so the reward $\mathcal{R}_i(a_i, c_i, c_{j\in\mathcal{S}})$ is informed by other confidence predictions $c_{j\in\mathcal{S}}$. **Right:** C@5 on unseen math benchmarks and on HotpotQA-to-OOD transfer.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

1 Introduction

Large language models are increasingly working in domains where they are far from perfectly accurate, yet can be trusted to solve a substantial fraction of the tasks they encounter. Confidence prediction allows these systems to identify the tasks that are hardest for the model, and defer them to a human expert or a stronger model, therefore increasing automation without sacrificing low risk. High stakes scenarios where automation is practical yet hardest tasks may be deferred to experts include visual question answering assistance for blind people [5, 25, 29], autonomous driving vehicles [38, 30, 3], or even coding agents, where orchestrators can decompose a problem, delegate and supervise work from specialized or cheaper worker agents [19, 2, 42, 11].

Autoregressive language models generate answers by sampling from next-token likelihoods, making sequence likelihood a natural source of a model's confidence in its own answer. However, common state-of-the-art post-training and inference techniques can severely degrade likelihood-based confidence as a predictor of answer correctness [41]. Specifically, Nakkiran et al. [41] show that reinforcement learning with verifiable rewards (RLVR) post-training [43, 17] moves models away from the likelihood-optimization mechanism that links likelihood to correctness by sharpening the distribution toward sequences rewarded by the verification mechanism, while chain-of-thought makes final-answer likelihood reflect consistency with a sampled reasoning path rather than uncertainty about the answer itself. Prior work estimates confidence with external signals or auxiliary modules: scoring rules built on model outputs [27, 24], probes on internal representations [4, 8], or selector models trained alongside the answerer [21, 37, 39, 52, 47]. More recently, LLMs themselves have been used for selective prediction by asking them to verbalize a confidence score alongside their answer [35, 33, 49, 55]. Without explicit training, models verbalize confidence poorly, so reinforcement learning that explicitly targets confidence prediction has therefore become the leading approach to obtain more reliable responses from models [15, 56, 53]. The rewards used to date score each confidence in isolation, pushing it toward 1 when the answer is correct and toward 0 when it is wrong [15, 56, 53]. At inference, confidence is used to select only the most confident answers to improve task coverage at a chosen risk. What matters is therefore how each value ranks relative to that threshold rather than where it lands on the unit interval. Risk tolerances vary across applications and evaluation settings (a coding orchestrator may accept a worker agent's lower-confidence patch when downstream tests will catch errors, whereas a VQA assistant for blind users or an autonomous vehicle can only act on high-confidence outputs), so the same confidence value can clear the abstention threshold under one operating point and fall short under another. A reward that scores each confidence in isolation therefore reduces the supervision to a push toward 0 or 1, and in practice produces polarized confidences, weak low-risk coverage, and poor generalization of selective prediction.

We propose to train confidence as a cross-question signal for which each per-completion target is conditioned on the other answers given by the model to the same prompt and other prompts, rather than only on its own correctness label. Selective prediction evaluates confidence over a collection of answers: for any chosen risk tolerance, a threshold determines which answers would be accepted and which would be abstained. We therefore reward each confidence by the selection behavior it would induce across the set, rather than by a fixed correctness target. Given a set of recent question-answer pairs with their confidence scores and correctness labels, we sort the set by confidence and identify, at a chosen risk tolerance, the largest subset of top-confidence answers whose error rate stays within tolerance. The confidence at the boundary of this acceptable subset defines the abstention threshold for that risk level, and each completion is then rewarded according to where its own confidence sits relative to this threshold: lying above is reinforced for correct answers, and lying below is reinforced for incorrect ones. Each risk tolerance induces its own threshold, so when reinforcing abstention at different risk levels, the same confidence value receives different rewards per risk level. Because every threshold depends on the set, the supervision a model gets on any one answer reflects how that answer compares to its peers rather than any fixed correctness target. Furthermore, we find that encouraging the model to predict diverse confidence scores, rather than collapsing around round numbers (e.g. single-decimal scores such as 0.8 or 0.9, or centesimals ending only in 0 or 5), improves coverage at low risk tolerances by preserving higher confidence-score resolution. Notably, this signal can be added to the RLVR post-training phase in LLM training [43, 17] at no extra cost: no separate confidence model or training stage, only additional rewards in the existing RL objective.

We evaluate CARE in two training regimes: math (Big-Math-Digits) and multi-hop Wikipedia QA (HotpotQA). CARE substantially improves coverage at low risk, both within the training domain

and on transfer to other domains, while maintaining competitive or even improving answer accuracy. Compared with RLCR, coverage at 5% risk rises from 0.8% to 42.4% on Math ID and from 12.3% to 23.6% on HotpotQA OOD. These improvements extend across risk levels, yielding lower area under the risk-coverage curve (AURC).

Contributions. We summarize our contributions as follows:

- We propose a cross-question coverage reward to train an LLM to verbalize confidence in its own answers, conditioning each completion's reward on abstention thresholds estimated from a set of answers rather than only its own binary correctness label, to maximize the number of questions answered within a user-specified risk tolerance.
- We introduce rewards that encourage diverse confidence scores across answers, preventing collapse onto a small set of values and thereby preserving the resolution needed for fine-grained ranking and more precise abstention decisions at low risks.
- We show that keeping confidence learning on-policy matters more than increasing the set of answers used to estimate the coverage reward: larger sets can reduce estimation noise, but accumulating answers from older rollouts makes the learning signal off-policy.

2 Related Work

Selective prediction. Selective prediction tries to maximize the utility of models (e.g. their accuracy) under some risk requirements, by only solving the subset of tasks the model is most confident about [12]. Approaches which directly learn to abstain [14, 21, 54] cannot adapt to the user's risk tolerance, which might be more strict in high-stakes scenarios. In contrast, a confidence score assigned to each answer can be used to satisfy various risk tolerances with a flexible abstention policy. If available, the model's estimated likelihood of the chosen label is an intuitive candidate for such a confidence score. For instance, for classification tasks, the predicted class' probability can be readily used to decide when to abstain [20]. Recently, similar approaches have been extended to large language models [59, 10]. However, due to their autoregressive nature, it is less clear which likelihood should represent certainty in the final answer. The likelihood of a full chain-of-thought includes many intermediate tokens unrelated to the final answer, while the likelihood of the final tokens is heavily conditioned on the preceding reasoning trace [41]. Post-training phases which give rise to LLMs' best capabilities, like reinforcement learning with verifiable rewards (RLVR) [43, 17], weaken likelihood as a confidence signal [41]. An alternative line of work uses an external selector to output confidence scores from the question, answer, supporting evidence, or internal representations such as embeddings or logits [52, 16, 40, 47]. These approaches duplicate knowledge across the answerer and the selector, add inference cost, and often require internal representations that are unavailable for stronger proprietary models. In contrast, we teach the answering model to reason about confidence in its own answer, simplifying inference to a single model that can already use its task knowledge.

Verbalized confidence prediction. Given the limitations of using an external selector or model intrinsics for confidence prediction, recent methods ask LLMs to verbalize confidence by outputting a numerical score in language tokens alongside each answer [49, 55]. LLMs' zero-shot confidence prediction performance is limited, so Damani et al. [15] propose learning confidence directly during the RLVR phase through an additional reward at no extra cost. More concretely, RLCR [15] trains LLMs' verbalized confidence with a Brier-style reward, equivalent to a mean-squared error between the confidence score and the binary correctness label, so each score is pushed toward 1 for a correct answer and 0 for an incorrect one. A second line of work keeps this pointwise structure but reshapes the reward to protect accuracy. Xuan et al. [56] propose CAR (Calibration Agentic RL), which adds a margin-separated reward to RLCR's Brier reward that preserves a gap between correct and incorrect answers independent of predicted confidence, in order to avoid reward hacking. Wu et al. [53] (BC-Beta) recast RLCR's Brier reward as integrating abstention behavior uniformly over all possible risk thresholds, and introduce a prior over user risk tolerances using a symmetric Beta distribution that up-weights both very-low-risk (high-stakes) and very-high-risk (high-coverage) regimes. These methods therefore target accuracy within the pointwise confidence-learning setting rather than changing the structure of the confidence signal itself. In contrast, CARE optimizes confidence prediction using a set of predicted answer-confidence pairs, with a new reward that

induces better coverage at risk rather than matching individual correctness. Because the two axes are complementary, accuracy-oriented refinements such as CAR's margin term or BC-Beta's risk prior could in principle be combined with our cross-question coverage rewards.

Learning from other samples. Several methods exploit information from other samples during learning: transductive learning [50, 31], test-time adaptation [26, 60], and learning to rank, where pairs or lists of examples provide supervision for their relative ordering [7, 9]. Concurrently to our work, RLSR [36] and ORCE [34] propose using other completions to compute confidence-related rewards. RLSR optimizes an AURC-based reward that depends only on confidence rankings, unlike CARE, which also uses confidence values to provide a stronger signal. ORCE trains a separate confidence model using DPO, with preferences determined by each prediction's contribution to the correlation between confidence rankings and correctness rates. Measuring an individual prediction's contribution to this set-wide statistic dilutes the learning signal. In contrast, CARE introduces a confidence reward that has a much stronger dependence on the correctness and confidence prediction of the completion being rewarded, while still being conditioned on the cross-question set of completions.

3 Cross-Question Abstention Reward Estimation (CARE)

Obtaining sensible confidence scores associated with a model's answers, and hence introducing a mechanism to abstain or defer to a human or model expert, can help maximize the share of questions answered automatically while keeping the risk of incorrect answers low. However, the ability of LLMs to do this zero-shot is limited, both through their intrinsic likelihoods or through verbalized confidence, especially when combined with chain-of-thought reasoning and post-training [41], see Section 2. Recent methods therefore train LLMs to express the confidence in their own answer outputting scores using standard tokens during the RLVR phase [15, 56, 53]. However, existing objectives reward confidence against binary correctness labels, rather than directly targeting coverage at the user's risk tolerance. CARE, in contrast, optimizes confidence prediction over a set of answers and confidence scores, enhancing their relative scores rather than optimizing each toward a binary target. Specifically, a *coverage reward* (Section 3.1) computes the abstention threshold from a set of question-answer pairs at a chosen risk tolerance and reinforces each confidence on the selection-preserving side. A *diversity-inducing reward* (Section 3.2) mechanism encourages the model to predict fine-grained confidence values and prevents collapse to coarse scores that are more natural in language like using single digits (e.g. 0.8 or 0.9) or multiples of 5 (e.g. 0.15 or 0.10).

3.1 Reinforcing coverage at a chosen risk tolerance

We first define coverage at risk, then use its confidence threshold to construct the training reward.

Coverage at risk (C@r). Given N answers, each with confidence score c_i and correctness label y_i, coverage at risk (C@r) measures the largest fraction that can be accepted while keeping its empirical error rate at most r. For an abstention threshold τ, coverage and selective error rate are defined as

$$C(\tau) = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}[c_i \geq \tau], \qquad R(\tau) = \frac{\sum_{i=1}^{N}(1-y_i)\,\mathbf{1}[c_i \geq \tau]}{\sum_{i=1}^{N}\mathbf{1}[c_i \geq \tau]}, \tag{1}$$

where $y_i \in \{0, 1\}$ indicates whether the answer is correct. We define coverage at user-defined risk r as the maximum coverage attained while the selective error rate $R(\tau)$ does not surpass the user's risk tolerance r. More precisely,

$$\mathrm{C@}r = \max_{\tau:\, R(\tau)\leq r} C(\tau), \qquad \tau^\star(r) \in \arg\max_{\tau:\, R(\tau)\leq r} C(\tau). \tag{2}$$

Coverage reward formulation. At training, we introduce an objective that aligns with C@r. Using a set of question-answer pairs $\mathcal{S} = \{(c_i, y_i)\}_{i=1}^{N}$, represented by their confidence scores and correctness labels, we estimate the risk-constrained threshold $\tau^\star(r)$. We then reward each completion according to where its own confidence sits relative to this threshold (Figure 2). Concretely,

$$\mathcal{R}^{\mathrm{C@}r} = (2y-1)\,\mathrm{sign}(c-\tau^\star(r))\,\big|c-\tau^\star(r)\big|^p, \tag{3}$$

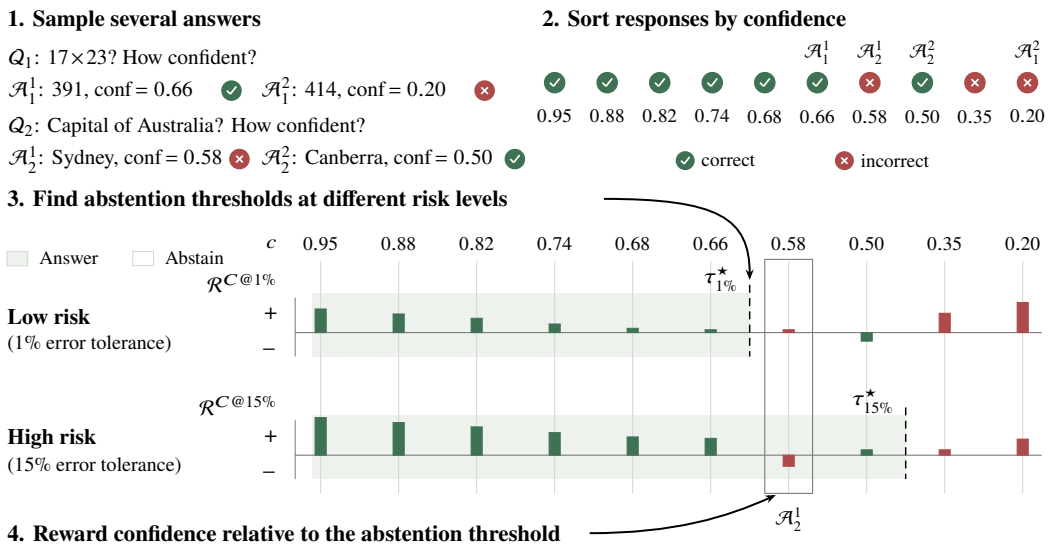


Figure 2: **CARE rewards confidence relative to abstention thresholds estimated across sets of questions and answers. 1.** Sample several answers and confidence scores for each question. **2.** Pool the answers across questions and sort them by confidence. **3.** For several error tolerances, find the confidence thresholds that maximize coverage while keeping the accepted answers' error rate within tolerance. **4.** Reward correct answers above the threshold and incorrect answers below it, and penalize the opposite placements, with reward magnitude increasing with distance from the threshold. The highlighted incorrect answer illustrates how the same confidence score can receive a positive reward under a stricter error tolerance and a negative reward under a more permissive one.

so correct answers receive positive reward above $\tau^\star(r)$ and negative reward below it, while incorrect answers receive the opposite signs. At the threshold the reward is zero. For example, at $\tau^\star = 0.5$ and $p = 2$, a correct answer at $c = 0.7$ receives $+0.04$, while one at $c = 0.3$ receives -0.04. p is a hyperparameter which highlights scores closer or further away from the abstention threshold, see Table 13. Thus this reward does not reinforce whether c matches the answer's correctness label in isolation, but whether c places the completion on the correct side of the abstention threshold that optimizes coverage at risk r. Because $\tau^\star(r)$ is recomputed from the model's current set of answers, the threshold reflects the model's achievable coverage at that point in training. Training over multiple risk tolerances exposes the same confidence distribution to several abstention thresholds, encouraging scores that support selection across potential downstream risk tolerances rather than only at one threshold, and hence providing a more fine-grained training signal. The size and scope of $\mathcal{S}$ trade off on-policy estimates against more robust abstention thresholds, as studied in Section 5.2.

3.2 Reinforcing diverse confidence scores

To enhance correct relative ordering of answers based on confidence, we want the model to use varied confidence scores. Concretely, coverage at low risk depends on fine distinctions among high-confidence answers that are not represented if completions collapse onto coarse scores such as 0.8 or 0.9. We therefore add two diversity-inducing rewards, one operating on the coarse shape of the confidence histogram and one on individual values. We study their contribution in Section 5.2.

Entropy reward. This reward encourages confidence mass to spread across coarse regions of $[0, 1]$. We partition $[0, 1]$ into B equal-width bins, where bin j covers $[(j-1)/B, j/B)$ for $j < B$, and the final bin includes 1. Let $j(c)$ denote the index of the bin containing confidence c. The normalized mass of bin j over the set $\mathcal{S}$ is

$$m_j = \frac{1}{|\mathcal{S}|} \sum_{(c_i, y_i) \in \mathcal{S}} \mathbf{1}[j(c_i) = j]. \tag{4}$$

Table 1: **Selective prediction training on math (in %):** ID (Big-Math-Digits, MATH-500, GSM8K), OOD (general knowledge & reasoning: CommonsenseQA, GPQA, SimpleQA, TriviaQA, HotpotQA). RLVR-Classifier is the only method not using verbalized confidence prediction from an LLM. **Bold** and underlining mark the best and second-best results among methods with verbalized confidence.

		ID					OOD				
Model	**Init**	**C@0**	**C@1**	**C@5**	**AURC↓**	**Acc**	**C@0**	**C@1**	**C@5**	**AURC↓**	**Acc**
RLVR-Classifier [15]	N/A	4.6	5.9	33.6	12.8	73.7	0.5	0.5	1.5	45.2	52.5
CAR-WeightedBrier [56]	Cold	0.0	0.0	0.0	20.6	71.8	0.1	0.1	0.1	41.7	51.3
CAR-MSCR [56]	Cold	0.0	0.0	0.0	19.2	72.0	0.1	0.1	7.0	37.7	56.7
BC-Beta [53]	Cold	0.0	0.0	0.0	17.6	71.8	0.0	0.0	11.6	**37.6**	55.7
RLSR [36]	Cold	0.0	0.0	0.0	21.7	72.7	0.0	0.0	0.0	51.0	43.9
ORCE-SC [34]	Cold	0.0	0.0	0.0	23.6	**75.0**	0.0	0.0	0.0	40.9	**56.8**
RLVR [15]	Cold	0.0	0.0	0.0	31.0	70.3	0.0	0.0	0.0	50.1	49.5
RLCR [15]	SFT	0.0	0.0	0.0	16.3	72.0	0.0	0.0	0.0	42.0	50.0
RLCR [15]	Cold	0.0	0.0	0.8	19.3	73.1	0.0	0.0	4.9	38.7	56.6
CARE (Ours)	Cold	**5.4**	**18.2**	**42.4**	**12.5**	73.1	**2.9**	**5.4**	**15.9**	37.7	53.6

The uniform target mass is $1/B$. For a completion with confidence c, the raw reward is

$$\tilde{\mathcal{R}}^{\text{ent}}(c) = 1 - B\, m_{j(c)}. \tag{5}$$

Thus underused bins receive positive reward, uniformly occupied bins receive zero reward, and overused bins receive negative reward. We clip the reward below at -1 to keep highly crowded bins from dominating, giving the entropy signal

$$\mathcal{R}^{\text{ent}}(c) = \max\{-1, \tilde{\mathcal{R}}^{\text{ent}}(c)\}. \tag{6}$$

Uniqueness reward. Whereas the entropy reward operates on coarse bins, models often default to a small set of specific confidence values. The uniqueness reward therefore acts directly on raw values, encouraging distinct confidence predictions within a set $\mathcal{S}$. With tolerance δ,

$$n^{\text{sim}}(c) = \sum_{(c_i, y_i) \in \mathcal{S}} \mathbf{1}[|c_i - c| \le \delta], \qquad \mathcal{R}^{\text{uniq}}(c) = 1 - 2\frac{n^{\text{sim}}(c)}{|\mathcal{S}|}. \tag{7}$$

Thus maximum $\mathcal{R}^{\text{uniq}}$ is assigned when the predicted confidence is unique in $\mathcal{S}$, with proportional reduction as more of the set repeats the same value.

3.3 Final objective

Building on existing confidence-prediction frameworks [15], we combine the coverage and diversity rewards with three per-completion signals carried over from prior work: a binary accuracy reward $\mathcal{R}^{\text{ans}}$ indicating whether the answer is correct, a format reward $\mathcal{R}^{\text{fmt}}$ that rewards emitting the answer and confidence in the expected output format, and a pointwise Brier confidence reward $\mathcal{R}^{\text{brier}} = 1 - (y - c)^2$:

$$\begin{aligned}\mathcal{R} = {} & w_{\text{fmt}}\, \mathcal{R}^{\text{fmt}} + w_{\text{ans}}\, \mathcal{R}^{\text{ans}} + w_{\text{brier}}\, \mathcal{R}^{\text{brier}} \\ & + \sum_{r \in \mathcal{T}} w_{\text{C@},r}\, \mathcal{R}^{\text{C@}r} + w_{\text{uniq}}\, \mathcal{R}^{\text{uniq}} + w_{\text{ent}}\, \mathcal{R}^{\text{ent}},\end{aligned} \tag{8}$$

where $\mathcal{T}$ is the set of risk tolerances included in training, and w represents the scalar weights, see Appendix A.1. We maintain the Brier reward for consistency with existing methods, and as a possible source of stable confidence-prediction learning in early stages of training, when coverage rewards can be noisier (Appendix A.6). In practice, we find that removing it has little or even positive impact, see Section 5.2.

4 Experiments

Training and benchmarks. We follow the training and evaluation protocol of Damani et al. [15] (RLCR), and consider two separate training settings: a *math* setting that trains on Big-Math-Digits [1],

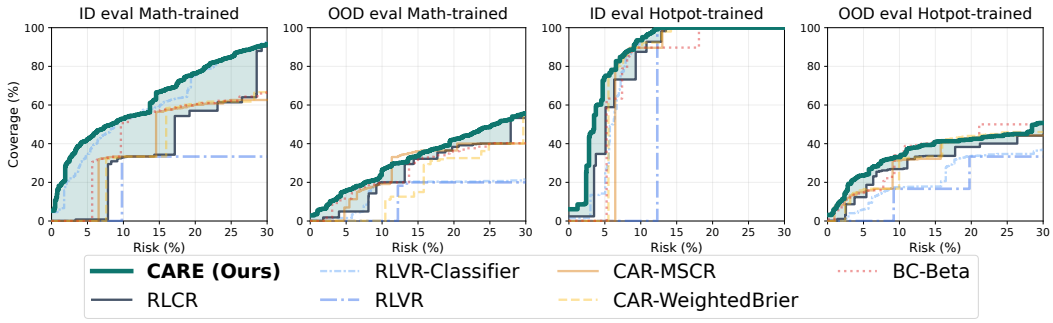


Figure 3: **CARE achieves higher coverage across most risk levels and tasks** on in-distribution (ID) and out-of-distribution (OOD) benchmarks, under both training regimes. CARE attains consistently higher coverage, with much smoother curves that allow coverage at any risk tolerance. Shading highlights the difference to RLCR, our most direct comparison.

and a *knowledge* setting that trains on HotpotQA [58]. In each setting, we train Qwen2.5-7B [57] with GRPO [46] and evaluate transfer to a suite of unseen benchmarks (Appendix Table 6). For math, these are MATH-500 [28] and GSM8K [13], and CommonsenseQA [48], GPQA [45], SimpleQA [51], TriviaQA [32], and HotpotQA for general knowledge and reasoning. We also follow the same evaluation protocol as Damani et al. [15]. Damani et al. [15] use a short SFT warm start before RL on Big-Math-Digits, but not HotpotQA. We find that cold-start matches or exceeds it (see Table 1) and do not perform SFT for CARE or other baselines. Further details are in Appendix A.1.

Selective prediction metrics. The question is whether confidence is *useful* for selective prediction: how many inputs can the system answer while keeping the error rate among answered inputs bounded? We therefore report **coverage at risk** (C@r) [20, 21] and **area under the risk–coverage curve** (AURC) [22], for a more global view of coverage. Standard calibration metrics such as ECE or Brier score do not directly answer this question: for example, a selector emitting a constant near the base accuracy already attains acceptable ECE while not enabling selection of some answers over others. Nonetheless, we include a calibration study in Appendix A.3.

Baselines. We compare against eight baselines. **RLVR** [15] takes a model trained with GRPO on accuracy alone and prompts it zero-shot to verbalize a confidence score. **RLVR-Classifier** [15] is a separately finetuned selector that predicts answer correctness from RLVR generations. This is not entirely comparable to the other settings, as the capacity for answering and confidence prediction is now duplicated, and separate optimization focused only on confidence prediction can be performed. Similarly, at inference, this method is twice as expensive as both answering and confidence prediction models (i.e. classifier) must be run. **RLCR** [15] adds a pointwise Brier-style confidence reward to GRPO and is the closest direct comparison to CARE. **CAR-WeightedBrier** [56], **CAR-MSCR** [56], and **BC-Beta** [53] are recent variants of the pointwise confidence reward that aim to balance accuracy and coverage: CAR adds a reward gap between correct and incorrect answers, and BC-Beta places a prior over risk tolerances that up-weights the very-low-risk and very-high-risk regimes. We use the default hyperparameters ($\lambda = 1/3$, $\beta_1 = \beta_2 = 1$, and a truncated $\mathrm{Beta}(0, 0)$ prior, respectively). Concurrently to our work, RLSR [36] and ORCE [34] introduce confidence rewards computed from multiple completions. **RLSR** rewards correct answers and penalizes incorrect ones according to their confidence ranks using an AURC-based objective. ORCE rewards confidence predictions that improve the ranking of questions with higher success rate. We adapt these reward designs to our setup, denoting the ORCE adaptation as **ORCE-SC**. Appendix A.1 contains implementation details.

5 Results

Following [15], we report results for both training regimes math and HotpotQA, on in- and out-of-distribution benchmarks. Other risk levels and additional metrics are reported in Appendix A.3.

Table 2: **Selective prediction training on HotpotQA:** ID (HotpotQA) & OOD (General Knowledge, Reasoning, Math: MATH-500, GSM8K, CommonsenseQA, GPQA, SimpleQA, TriviaQA). RLVR-Classifier is the only method not using verbalized confidence prediction from an LLM. **Bold** and underlining mark the best and second-best results among methods with verbalized confidence.

	ID					OOD				
Model	**C@0**	**C@1**	**C@5**	**AURC↓**	**Acc**	**C@0**	**C@1**	**C@5**	**AURC↓**	**Acc**
RLVR-Classifier [15]	1.6	1.6	14.2	6.5	87.7	0.4	0.4	6.0	42.0	53.9
CAR-WeightedBrier [56]	0.0	0.0	0.0	6.4	85.8	0.0	0.0	16.5	35.6	57.3
CAR-MSCR [56]	0.0	0.0	0.0	7.2	87.0	0.0	**7.4**	14.8	36.2	56.0
BC-Beta [53]	0.1	0.1	0.1	7.2	81.9	0.9	5.6	14.8	**30.5**	**63.4**
RLSR [36]	0.0	0.0	0.0	8.6	86.6	0.0	0.0	14.5	37.7	58.8
ORCE-SC [34]	0.0	0.0	0.0	10.7	85.6	0.0	0.0	14.9	38.5	55.2
RLVR [15]	0.0	0.0	0.0	12.3	**87.7**	0.0	0.0	0.0	46.0	53.9
RLCR [15]	2.4	2.4	34.7	6.1	86.1	0.1	0.1	12.3	36.6	56.2
CARE (Ours)	**6.1**	**6.1**	**74.7**	**4.4**	87.6	**3.1**	6.4	**23.6**	33.2	58.0

5.1 Main results

CARE unlocks low-risk coverage. Across both training regimes and evaluation settings (Tables 1 and 2), CARE consistently unlocks coverage at low risk: C@0%, C@1%, and C@5% generally rise by a wide margin over RLVR, RLCR, as well as CAR and BC-Beta variants, and even the RLVR-Classifier. As can be seen in Figure 3, CARE provides much better coverage at low risks. This is the failure mode of pointwise training: a model can look calibrated on average while still failing to order its highest-confidence answers sharply enough for low-risk abstention. Furthermore, CARE improves coverage over RLSR and ORCE-SC, likely due to the more direct per-completion feedback provided by CARE. RLSR uses only the relative ranking between confidence scores, not numerical confidence differences as CARE does, providing sparser feedback. ORCE-SC measures each prediction's marginal contribution to a set-wide correlation, which dilutes the individual learning signal. In contrast, CARE provides more guidance, rewarding each completion based on the difference between its own confidence and the abstention threshold computed from the set of completions.

CARE improves coverage beyond highest stakes. Figure 3 shows that, even though we optimize specific risk levels during training, the coverage curve progresses smoothly with increased risk, unlike existing methods. CARE improves AURC over RLCR in all training and evaluation scenarios, demonstrating that the improvements in coverage are not limited to the highest stakes. The only case where an alternative method can compete with CARE in terms of AURC is OOD evaluation, where BC-Beta achieves higher accuracy and therefore provides better AURC. However, its confidence prediction is worse at low risk tolerance (Figure 3) and ID performance is worse overall, including accuracy (e.g. 87.6% for CARE on ID HotpotQA vs. 81.9% for BC-Beta). Combining an accuracy-maximizing objective (orthogonal to our contribution) with CARE could recover such behavior.

Accuracy is preserved or improved. Compared to its closest comparison, RLCR, CARE matches or improves accuracy across most splits, and substantially improves over RLVR, which trains with accuracy alone, on both ID and OOD averages. Adding cross-question coverage and diversity rewards, therefore, does not trade off task accuracy for confidence quality. CAR-WeightedBrier and BC-Beta target a different axis: they sharpen accuracy within the confidence-learning setting via strict-margin rewards that preserve a gap between correct and incorrect answers (CAR) or via priors over risk tolerances that emphasize the very-low-risk and very-high-risk regimes (BC-Beta). They sometimes attain higher accuracy than CARE, especially on some OOD non-math tasks, but generally remain inferior at strict risk thresholds. Their accuracy refinements act on a complementary axis to our cross-question coverage rewards and could, in principle, be combined with CARE.

Figure 4 shows histograms of predicted confidence per model, grouped on two coarseness levels. When looking at the coarser plots (top), we can see that the baselines (RLVR, RLVR-Classifier, and RLCR) concentrate on the high-confidence regime, hence often being over-confident, especially in the zero-shot RLVR case. CARE, in contrast, frequently predicts low confidence for answers

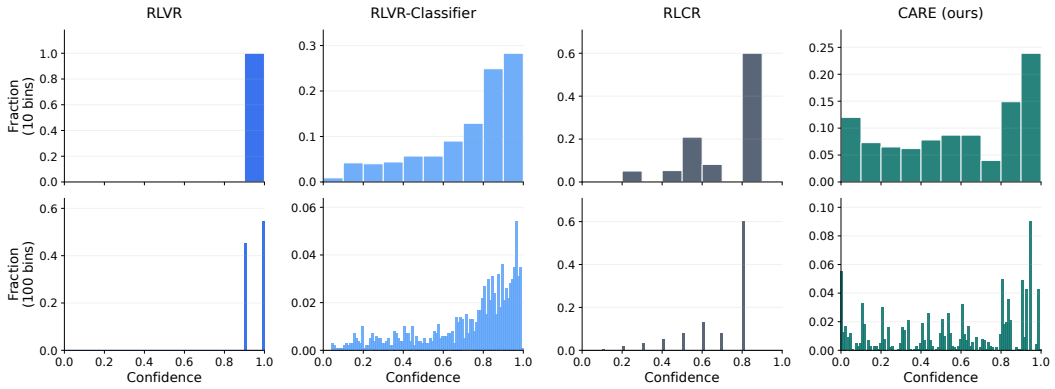


Figure 4: **Histogram of predicted confidence values** on Big-Math-Digits using 10 (top) and 100 (bottom) bins. Baselines are overconfident. Verbalized baselines collapse predictions to simple decimals. **CARE predicts fine-grained confidence scores, enabling coverage at very low risks; and assigning low confidence for incorrect answers, allowing abstention for difficult tasks.**

Table 3: **CARE improves confidence prediction for Gemma 4 E2B-it, showing its benefits generalize to other model families and sizes.** Training on math, with ID evaluation on Big-Math-Digits, MATH-500, and GSM8K, and OOD evaluation on CommonsenseQA, GPQA, SimpleQA, TriviaQA, and HotpotQA. All metrics are percentages. **Bold** marks the best result.

	ID						OOD					
Method	**C@0**	**C@1**	**C@5**	**C@10**	**AURC↓**	**Acc**	**C@0**	**C@1**	**C@5**	**C@10**	**AURC↓**	**Acc**
RLCR [15]	0.0	0.0	0.0	32.8	17.8	74.2	0.0	0.0	11.4	33.7	37.6	53.3
CARE (Ours)	**1.7**	**1.7**	**17.9**	**50.2**	**13.7**	**74.9**	**1.2**	**1.2**	**23.4**	**35.9**	**36.2**	**55.5**

it is unsure about. We can also look at the more fine-grained histogram at the bottom of Figure 4. When looking at the verbalized confidence prediction baselines, we observe that only single-digit values are used to predict confidence. Even though confidence optimization introduced by RLCR optimizes the model away from the degenerate overconfidence of vanilla RLVR, we see that it does not induce diverse confidence scores. In Appendix A.4 we show how the other brier-like confidence optimization baselines also follow this pattern. Only the RLVR-Classifier, which utilizes a linear head and hence a more natural continuous output space, and CARE utilize a wider range of scores. CARE still frequently predicts round numbers with fewer decimals, but also uses many other values.

Results on another model family and size. We also train Gemma 4 E2B-it [23], a smaller model with 2.3B effective parameters from a different family than Qwen2.5-7B, on Big-Math-Digits. We compare against RLCR, our most direct comparison, to test whether the gains from cross-question coverage and diversity rewards extend beyond Qwen2.5-7B. Compared with RLCR, CARE improves C@5 from 0.0% to 17.9% on ID and from 11.4% to 23.4% on OOD, as well as AURC and accuracy (Table 3). This confirms the applicability of CARE to a more recent model family at a smaller scale.

5.2 Ablations

Reward design ablation. Table 4 ablates the reward terms in our final objective when training in the math setting. We observe that our design is robust to removing Brier reward (row 2 vs. row 1), which causes little or no harm, obtaining slightly better AURC and coverage at some risks. We observe similar robustness when optimizing only the strictest risk level, C@0, instead of multiple risk levels (row 3 vs. row 1). This is overall slightly worse, but generally better than existing methods. However, removing our coverage rewards (row 4) is significantly damaging, highlighting the importance of our across-question coverage reward for good selective prediction. This shows diversity encouraging rewards are not enough, brier reward alone does not encourage correct relative confidence scores. Dropping $\mathcal{R}^{uniq}$ (row 5) erodes C@5 (42.4 $\rightarrow$ 30.2), dropping $\mathcal{R}^{ent}$ (row 6) erodes it further (42.4

Table 4: **Reward-design ablations** in the math training regime. Row 1 (#1) is the full CARE recipe. Removing Brier (#2) largely preserves and even improves performance, whereas removing the coverage reward introduced by CARE (#4) hurts performance the most. Diversity rewards are useful and complementary (#5,#6), but not enough without the coverage reward (#4). Optimizing coverage at a single risk level (#3) is worse than at multiple (#1), but better than the baselines (#4).

	Reward terms					ID						OOD					
#	$\mathcal{R}^{\text{brier}}$	$\mathcal{R}^{\text{C@0}}$	$\mathcal{R}^{\text{C@}r}$	$\mathcal{R}^{\text{uniq}}$	$\mathcal{R}^{\text{ent}}$	**C@0**	**C@1**	**C@5**	**C@10**	**AURC↓**	**Acc**	**C@0**	**C@1**	**C@5**	**C@10**	**AURC↓**	**Acc**
1	✓	✓	✓	✓	✓	**5.4**	**18.2**	**42.4**	52.9	12.5	73.1	**2.9**	5.4	15.9	26.4	37.7	53.6
2	×	✓	✓	✓	✓	3.0	3.0	40.3	**55.8**	**12.1**	**73.9**	2.6	4.0	16.1	22.1	36.9	56.5
3	✓	✓	×	✓	✓	2.2	3.0	33.1	54.2	12.7	72.3	0.5	2.5	14.4	28.7	36.8	53.7
4	✓	×	×	✓	✓	0.6	0.6	16.3	33.3	19.3	72.4	0.4	0.4	4.6	20.9	41.5	55.7
5	✓	✓	✓	×	✓	3.6	3.6	30.2	51.6	13.2	72.3	2.0	**6.9**	**18.5**	26.3	**33.5**	59.7
6	✓	✓	✓	✓	×	0.0	0.0	20.1	37.5	14.7	73.2	0.0	0.0	13.7	**31.0**	34.3	**60.2**

Table 5: **Analysis of the effect of the scope and size of the set of answers across which we compute rewards.** In the math training regime, we study the tradeoff between smaller on-policy sets and larger sets that provide more robust abstention threshold estimates.

		ID Math						OOD General Reasoning and Knowledge					
Scope	**Size**	**C@0**	**C@1**	**C@5**	**C@10**	**AURC↓**	**Acc**	**C@0**	**C@1**	**C@5**	**C@10**	**AURC↓**	**Acc**
Per-group	32	0.0	0.0	0.0	33.3	19.9	72.6	0.0	0.0	0.0	3.8	46.3	46.2
Per-worker	192	**5.4**	**18.2**	**42.4**	52.9	**12.5**	73.1	**2.9**	**5.4**	15.9	26.4	37.7	53.6
Across-workers	1,152	2.2	2.2	28.7	**54.3**	13.3	71.9	1.5	1.5	**18.2**	**31.0**	**34.3**	57.5
Rolling buffer	5,000	0.0	0.0	0.0	33.3	24.3	**73.7**	0.0	0.0	0.0	0.0	42.0	57.7
Rolling buffer	10,000	0.0	0.0	0.0	33.3	33.9	73.3	0.3	0.3	0.8	20.1	41.0	**59.6**

$\rightarrow$ 20.1). Hence, diversity rewards are helpful and not redundant: the coarse-bin entropy term and the value-level uniqueness term each help enhance the resolution of confidence scores.

Size of the set for cross-question rewards: a tradeoff between on-policy signal and robust risk estimation. The coverage and diversity rewards are computed against a set $\mathcal{S}$ of (c_i, y_i) pairs (Section 3.1). Table 5 varies the size and scope of this set, ranging from a single GRPO group (32 completions) up to a 10,000-sample rolling buffer. Increasing the size can yield more robust estimates of the abstention threshold, helping to avoid noisy rewards. However, a larger scope quickly becomes off-policy, as answers and confidence scores are generated by older versions of the model, which do not represent its current knowledge and confidence prediction capabilities. For math, following [15], we use a GRPO group size of 32 (i.e. sample 32 answers to each question to estimate the baseline value), with 6 questions per GPU worker and 6 GPU workers in total. Group-level pools (32) are too small to give a stable abstention threshold, and thus don't achieve coverage at low risk. The per-worker and across-worker groups are the best (192 and 1,152 completions respectively). This represents the sweet spot, as they are on-policy yet provide sound abstention thresholds during training. Per-worker is more efficient and achieves similar performance (higher ID but lower OOD) compared to across-workers, so we use that for the rest of our experiments. One can accumulate a buffer of answer correctness labels and confidence scores from past updates. However, larger sets perform poorly, likely due to off-policiness being too damaging and the added robustness small.

6 Conclusion

An LLM aware of the limits of its own knowledge and abilities can operate reliably in an autonomous manner. We propose CARE, a reward added to RLVR that teaches the model to verbalize this confidence. Unlike prior work that rewards each confidence against its own binary correctness label, CARE aggregates predicted confidences across a batch and reinforces each one relative to the others. We demonstrate empirically across two training settings and various math, general knowledge, and reasoning benchmarks that CARE improves selective prediction, unlocking coverage at low risk without sacrificing overall accuracy. Concretely, it enables a coverage at 5% risk on math benchmarks of 42% when existing verbalized baselines cannot surpass 1%. We hope this motivates further work on learning to predict confidence for reliable reasoning, allowing control over risk in high stakes environments, as the coverage at low risk is still low in absolute terms compared to overall accuracy.

Acknowledgments

This research was partially funded by an Alexander von Humboldt Professorship in Multimodal Reliable AI, sponsored by the Federal Ministry of Research, Technology, and Space (BMFTR), by a LOEWE-Spitzen-Professur (LOEWE/4a//519/05.00.002(0010)/93), and has benefited from the Excellence Cluster "Reasonable AI" by the German Research Foundation (Deutsche Forschungsgemeinschaft – DFG) under Germany's Excellence Strategy – EXC-3057. We gratefully acknowledge support from the hessian.AI Service Center (funded by the Federal Ministry of Research, Technology and Space, BMFTR, grant no. 16IS22091) and the hessian.AI Innovation Lab (funded by the Hessian Ministry for Digital Strategy and Innovation, grant no. S-DIW04/0013/003).

References

[1] Alon Albalak, Duy Phung, Nathan Lile, Rafael Rafailov, Kanishk Gandhi, Louis Castricato, Anikait Singh, Chase Blagden, Violet Xiang, Dakota Mahan, et al. Big-math: A large-scale, high-quality math dataset for reinforcement learning in language models. *arXiv preprint arXiv:2502.17387*, 2025.

[2] Anthropic. Subagents in Claude Code. `https://docs.anthropic.com/en/docs/claude-code/sub-agents`, 2026. Accessed: 2026-05-02.

[3] Pasquale Antonante, Sushant Veer, Karen Leung, Xinshuo Weng, Luca Carlone, and Marco Pavone. Task-aware risk estimation of perception failures for autonomous vehicles. In *Proceedings of Robotics: Science and Systems*, 2023.

[4] Amos Azaria and Tom Mitchell. The internal state of an LLM knows when it's lying. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023.

[5] Jeffrey P. Bigham, Chandrika Jayant, Hanjie Ji, Greg Little, Andrew Miller, Robert C. Miller, Robin Miller, Aubrey Tatarowicz, Brandyn White, Samuel White, and Tom Yeh. VizWiz: Nearly real-time answers to visual questions. In *Proceedings of the 23rd Annual ACM Symposium on User Interface Software and Technology*, 2010.

[6] Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950. doi: 10.1175/1520-0493(1950)078<0001:VOFEIT>2.0.CO;2.

[7] Christopher Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In *Proceedings of the 22nd International Conference on Machine Learning*, 2005.

[8] Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. Discovering latent knowledge in language models without supervision. In *The Eleventh International Conference on Learning Representations*, 2023.

[9] Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. Learning to rank: From pairwise approach to listwise approach. In *Proceedings of the 24th International Conference on Machine Learning*, 2007.

[10] Jiefeng Chen, Jinsung Yoon, Sayna Ebrahimi, Sercan Arik, Tomas Pfister, and Somesh Jha. Adaptation with self-evaluation to improve selective prediction in LLMs. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5190–5213, 2023. doi: 10.18653/v1/2023.findings-emnlp.345.

[11] Lingjiao Chen, Matei Zaharia, and James Zou. FrugalGPT: How to use large language models while reducing cost and improving performance. *arXiv preprint arXiv:2305.05176*, 2023.

[12] C. K. Chow. On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*, 16(1):41–46, 1970. doi: 10.1109/TIT.1970.1054406.

[13] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

[14] Corinna Cortes, Giulia DeSalvo, and Mehryar Mohri. Boosting with abstention. In *Advances in Neural Information Processing Systems*, 2016.

[15] 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 *International Conference on Learning Representations*, 2026.

[16] Corentin Dancette, Spencer Whitehead, Rishabh Maheshwary, Ramakrishna Vedantam, Stefan Scherer, Xinlei Chen, Matthieu Cord, and Marcus Rohrbach. Improving selective visual question answering by learning from your peers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24049–24059, 2023.

[17] DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, et al. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[18] Tom Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006. doi: 10.1016/j.patrec.2005.10.010.

[19] Adam Fourney, Gagan Bansal, Hussein Mozannar, Cheng Tan, Eduardo Salinas, Erkang Zhu, Friederike Niedtner, Grace Proebsting, Griffin Bassman, Jack Gerrits, Jacob Alber, Peter Chang, Ricky Loynd, Robert West, Victor Dibia, Ahmed Awadallah, Ece Kamar, Rafah Hosn, and Saleema Amershi. Magentic-One: A generalist multi-agent system for solving complex tasks. *arXiv preprint arXiv:2411.04468*, 2024.

[20] Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In *Advances in Neural Information Processing Systems*, 2017.

[21] Yonatan Geifman and Ran El-Yaniv. SelectiveNet: A deep neural network with an integrated reject option. In *Proceedings of the 36th International Conference on Machine Learning*, 2019.

[22] Yonatan Geifman, Guy Uziel, and Ran El-Yaniv. Bias-reduced uncertainty estimation for deep neural classifiers. In *International Conference on Learning Representations*, 2019.

[23] Gemma Team. Gemma 4 technical report, 2026. URL `https://arxiv.org/abs/2607.02770`.

[24] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, 2017.

[25] Danna Gurari, Qing Li, Abigale J. Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P. Bigham. VizWiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[26] Moritz Hardt and Yu Sun. Test-time training on nearest neighbors for large language models. In *International Conference on Learning Representations*, 2024.

[27] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations*, 2017.

[28] 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. *Advances in Neural Information Processing Systems*, 2021.

[29] Mina Huh, Fangyuan Xu, Yi-Hao Peng, Chongyan Chen, Hansika Murugu, Danna Gurari, Eunsol Choi, and Amy Pavel. Long-form answers to visual questions from blind and low vision people. In *Conference on Language Modeling*, 2024.

[30] Boris Ivanovic, Richard Lin, Shubham Shrivastava, Punarjay Chakravarty, and Marco Pavone. Propagating state uncertainty through trajectory forecasting. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 2022.

[31] Thorsten Joachims. Transductive inference for text classification using support vector machines. In *Proceedings of the 16th International Conference on Machine Learning*, 1999.

[32] Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 1601–1611, 2017.

[33] Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

[34] Chen Li, Xiaoling Hu, Songzhu Zheng, Jiawei Zhou, and Chao Chen. ORCE: Order-aware alignment of verbalized confidence in large language models. *arXiv preprint arXiv:2605.12446*, 2026. URL `https://arxiv.org/abs/2605.12446`.

[35] Stephanie Lin, Jacob Hilton, and Owain Evans. Teaching models to express their uncertainty in words. *arXiv preprint arXiv:2205.14334*, 2022.

[36] Gaoxiang Luo, Yifan Wu, Sinian Zhang, Aryan Deshwal, and Ju Sun. Aligning language models with selective prediction. *arXiv preprint arXiv:2607.03528*, 2026. URL `https://arxiv.org/abs/2607.03528`.

[37] David Madras, Toniann Pitassi, and Richard Zemel. Predict responsibly: Improving fairness and accuracy by learning to defer. In *Advances in Neural Information Processing Systems*, 2018.

[38] Rowan McAllister, Yarin Gal, Alex Kendall, Mark van der Wilk, Amar Shah, Roberto Cipolla, and Adrian Weller. Concrete problems for autonomous vehicle safety: Advantages of Bayesian deep learning. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, 2017.

[39] Hussein Mozannar and David Sontag. Consistent estimators for learning to defer to an expert. In *Proceedings of the 37th International Conference on Machine Learning*, 2020.

[40] Erum Mushtaq, Zalan Fabian, Yavuz Faruk Bakman, Anil Ramakrishna, Mahdi Soltanolkotabi, and Salman Avestimehr. HARMONY: Hidden activation representations and model output-aware uncertainty estimation for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2025.

[41] Preetum Nakkiran, Arwen Bradley, Adam Goliński, Eugene Ndiaye, Michael Kirchhof, and Sinead Williamson. Trained on tokens, calibrated on concepts: The emergence of semantic calibration in LLMs. In *International Conference on Learning Representations*, 2026.

[42] Nous Research. Subagent delegation. `https://hermes-agent.nousresearch.com/docs/user-guide/features/delegation`, 2026. Accessed: 2026-05-02.

[43] OpenAI. OpenAI o1 system card. `https://openai.com/index/openai-o1-system-card/`, 2024.

[44] John C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in Large Margin Classifiers*, pages 61–74. MIT Press, 1999.

[45] David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level google-proof Q&A benchmark. In *Conference on Language Modeling*, 2024.

[46] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

[47] Tejas Srinivasan, Jack Hessel, Tanmay Gupta, Bill Yuchen Lin, Yejin Choi, Jesse Thomason, and Khyathi Chandu. Selective "selective prediction": Reducing unnecessary abstention in vision-language reasoning. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 12935–12948, 2024. doi: 10.18653/v1/2024.findings-acl.767.

[48] Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4149–4158, 2019.

[49] Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D. Manning. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

[50] Vladimir N. Vapnik. *Statistical Learning Theory*. Wiley, 1998.

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

[52] Spencer Whitehead, Suzanne Petryk, Vedaad Shakib, Joseph Gonzalez, Trevor Darrell, Anna Rohrbach, and Marcus Rohrbach. Reliable visual question answering: Abstain rather than answer incorrectly. In *Computer Vision – ECCV 2022*, pages 148–166. Springer, 2022. doi: 10.1007/978-3-031-20059-5_9.

[53] 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.

[54] Ji Xin, Raphael Tang, Yaoliang Yu, and Jimmy Lin. The art of abstention: Selective prediction and error regularization for natural language processing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*, 2021. doi: 10.18653/v1/2021.acl-long.84.

[55] Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. Can LLMs express their uncertainty? an empirical evaluation of confidence elicitation in LLMs. In *The Twelfth International Conference on Learning Representations*, 2024.

[56] Weihao Xuan, Qingcheng Zeng, Heli Qi, Yunze Xiao, Junjue Wang, and Naoto Yokoya. The confidence dichotomy: Analyzing and mitigating miscalibration in tool-use agents. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11325–11349, 2026. doi: 10.18653/v1/2026.acl-long.520.

[57] An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

[58] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, 2018.

[59] Hiyori Yoshikawa and Naoaki Okazaki. Selective-LAMA: Selective prediction for confidence-aware evaluation of language models. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 2017–2028, 2023. doi: 10.18653/v1/2023.findings-eacl.150.

[60] Chenyu Zhang, Kunlun Xu, Zichen Liu, Yuxin Peng, and Jiahuan Zhou. SCAP: Transductive test-time adaptation via supportive clique-based attribute prompting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

A Supplementary Material

Table 6: Training datasets and evaluation benchmarks. Further details in Appendix A.1.

Dataset	Topic	Example
Training Datasets		
Big-Math-Digits	Mixed-domain math	*Q:* Given $X \sim B(n, 0.8)$ and $D(X) = 1.6$, find n. *A:* 10
HotpotQA	Wikipedia multi-hop QA	*Q:* Were Scott Derrickson and Ed Wood of the same nationality? *A:* yes
Test Benchmarks		
MATH-500	Competition math	*Q:* Convert $(0, 3)$ from rectangular to polar coordinates. *A:* $\left(3, \frac{\pi}{2}\right)$
GSM8K	Grade-school math	*Q:* Janet's ducks lay 16 eggs; after eating 3 and baking with 4, how much money does she make selling the rest at \$2 each? *A:* 18
CommonsenseQA	Commonsense reasoning	*Q:* A revolving door serves as security at a _? (A) bank ... *A:* bank
GPQA	Graduate science	*Q:* Two quantum states have lifetimes 10^{-9} s and 10^{-8} s; what resolvable energy gap fits? *A:* 10^{-4} eV
SimpleQA	Short factual QA	*Q:* Who received the IEEE Frank Rosenblatt Award in 2010? *A:* Michio Sugeno
TriviaQA	Trivia QA	*Q:* Which Lloyd Webber musical features the Phantom? *A:* The Phantom of the Opera

A.1 Additional experimental details

Training configuration. We follow the training and evaluation protocol of Damani et al. [15] (RLCR) unless stated otherwise. We train Qwen2.5-7B with GRPO using the same answer-plus-confidence prompt format as RLCR. The selected CARE runs use the per-worker reward scope: each GPU worker computes the cross-question rewards over its local batch of 6 prompts (8 for HotpotQA) with 32 completions each, for 192 completions (256 for HotpotQA) per reward pool. Invalidly formatted completions are excluded from the confidence/correctness pool. The coverage rewards use risk tolerances $\mathcal{T} = \{0, 0.01, 0.05, 0.10, 0.15, 0.20, 0.25\}$ and power $p = 2$. The entropy reward uses 100 equal-width bins on $[0, 1]$, and the uniqueness reward uses tolerance $\delta = 0.01$, so confidence values within one percentage point are treated as similar. Before weighting, we square the clipped entropy reward while preserving its sign, $u \mapsto u|u|$. We ablate using instead or in addition a 10-bin entropy reward and obtain worse results than with the 100-bin version. These provide different diversity encouragements around bin boundaries even with similar granularity, which is relevant when the model tends to predict identical scores. The selected reward weights are 0.5 for format, 0.5 for answer accuracy, 0.1 for Brier, 0.1 for uniqueness, 0.1 for entropy, and 0.1 total for the coverage rewards, split equally across risk levels. For evaluation, tied confidence scores are accepted together, without using correctness to break ties. We find that to avoid reward hacking, maintaining a balance between accuracy reward weight and other confidence related reward weights is quite relevant, increasing the coverage rewards carelessly can lead to scenarios where the model learns to give terrible answers with confidence 0, hence obtaining optimal confidence scores.

Answer extraction and judging. Final answers are parsed from the standard RLCR answer-plus-confidence format. Math benchmarks (Big-Math-Digits, MATH-500, GSM8K) are scored with the symbolic checker from Damani et al. [15]. The knowledge suite (CommonsenseQA, GPQA, SimpleQA, TriviaQA, HotpotQA) is scored with the LLM-as-judge pipeline of Damani et al. [15], using the same judge model and prompt.

Baselines. For **RLVR**, **RLVR-Classifier**, and **RLCR**, we use the public checkpoints released by Damani et al. [15]. We implement the baselines for **CAR-WeightedBrier**, **CAR-MSCR**, and **BC-Beta** and follow the same training setting for these. For **RLSR** [36], we introduce their AURC-based reward into our setting, using full-parameter training and the same learning rate as all other methods. For ORCE [34], we introduce their Spearman-correlation reward into our GRPO training, instead of training a separate confidence model with SFT and DPO as in the original work, as we seek to compare the reward designs in isolation. We denote this adaptation as **ORCE-SC**.

Table 7: **Math, in-distribution** (Big-Math-Digits, 1,000 examples, `math-verify`).

Model	Init	C@0	C@1	C@5	C@10	C@15	C@20	C@25	C@30	AURC↓	Acc
RLVR-Classifier [15]	N/A	8.2	12.1	25.5	49.0	65.1	80.1	89.2	99.2	11.9	69.8
CAR-WeightedBrier [56]	Cold	0.0	0.0	0.0	0.0	0.0	81.4	85.3	**100.0**	18.5	71.3
CAR-MSCR [56]	Cold	**0.1**	**0.1**	0.1	0.1	69.5	80.4	87.8	87.8	17.5	68.8
BC-Beta [53]	Cold	0.0	0.0	0.0	51.9	71.6	81.5	86.6	**100.0**	13.9	70.0
RLSR [36]	Cold	0.0	0.0	0.0	0.0	0.0	83.7	85.6	**100.0**	20.0	70.7
ORCE-SC [34]	Cold	0.0	0.0	0.0	0.0	0.0	0.0	0.0	**100.0**	26.2	71.6
RLVR [15]	Cold	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	34.7	66.2
RLCR [15]	SFT	0.0	0.0	0.0	0.0	59.0	74.0	86.7	**100.0**	15.9	70.4
RLCR [15]	Cold	**0.1**	**0.1**	0.1	0.1	0.1	68.3	81.4	97.6	20.1	68.7
CARE (Ours)	Cold	**0.1**	**0.1**	**34.5**	**58.4**	**74.2**	**85.0**	**94.8**	**100.0**	**10.7**	**72.4**

Table 8: **Math-trained, out-of-distribution average** over CommonsenseQA, GPQA, SimpleQA, TriviaQA, and HotpotQA, scored with the LLM judge.

Model	Init	C@0	C@1	C@5	C@10	C@15	C@20	C@25	C@30	AURC↓	Acc
RLVR-Classifier [15]	N/A	0.5	0.5	1.5	19.7	20.4	20.4	20.7	21.6	45.2	52.5
CAR-WeightedBrier [56]	Cold	0.1	0.1	0.1	0.1	15.0	32.5	39.9	55.0	41.7	51.3
CAR-MSCR [56]	Cold	0.1	0.1	7.0	19.1	**36.1**	36.5	40.1	40.1	37.7	56.7
BC-Beta [53]	Cold	0.0	0.0	11.6	19.6	32.3	36.1	39.8	40.3	**37.6**	55.7
RLSR [36]	Cold	0.0	0.0	0.0	0.0	0.0	0.0	13.0	13.0	51.0	43.9
ORCE-SC [34]	Cold	0.0	0.0	0.0	20.0	20.0	20.0	38.1	40.0	40.9	**56.8**
RLVR [15]	Cold	0.0	0.0	0.0	0.0	20.0	20.0	20.0	20.0	50.1	49.5
RLCR [15]	SFT	0.0	0.0	0.0	7.5	19.6	32.5	38.6	51.2	42.0	50.0
RLCR [15]	Cold	0.0	0.0	4.9	20.0	32.2	38.3	40.1	53.3	38.7	56.6
CARE (Ours)	Cold	**2.9**	**5.4**	**15.9**	**26.4**	33.8	**42.1**	**48.6**	**55.6**	37.7	53.6

A.2 Compute resources

Training was run on NVIDIA H100 and A100 80GB GPUs. Across the reported training settings, each run completed in under 24 hours.

A.3 Additional metrics

Additional coverage levels. Tables 7 to 9 report coverage across the full set of evaluated risk levels for the math-trained models, and Tables 10 and 11 do the same for the HotpotQA-trained models. Some entries remain zero even at high nominal risk levels because coverage is computed by thresholding the model's emitted confidence values, so a model with only a few tied confidence scores has only a few possible operating points. For example, on Big-Math-Digits, RLVR emits only four confidence values. The accepted sets obtained by thresholding these scores have error rates between 33.8% and 35.7%, so none satisfies the evaluated risk tolerances of up to 30%. Consequently, its coverage remains zero at all reported risk levels.

Additional ranking and calibration metrics. We complement coverage at different risk levels and AURC with AUROC, expected calibration error (ECE), and Brier score.

Confidence ranking. AUROC measures whether correct answers receive higher confidence than incorrect answers, considering all correct–incorrect pairs [18]. Let I_1 and I_0 denote the sets of correct and incorrect completions:

$$\mathrm{AUROC} = \frac{1}{|I_1||I_0|} \sum_{i \in I_1} \sum_{j \in I_0} \left(\mathbf{1}[c_i > c_j] + \tfrac{1}{2}\mathbf{1}[c_i = c_j] \right). \tag{9}$$

Each pair contributes equally. In contrast, AURC evaluates the error rates of accepted sets obtained by varying a confidence threshold, giving greater importance to errors among the highest-confidence

Table 9: **Math average** over Big-Math-Digits, MATH-500, and GSM8K.

Model	Init	C@0	C@1	C@5	C@10	C@15	C@20	C@25	C@30	AURC↓	Acc
RLVR-Classifier [15]	N/A	4.6	5.9	33.6	52.7	59.2	75.4	85.5	92.7	12.8	73.7
CAR-WeightedBrier [56]	Cold	0.0	0.0	0.0	33.3	33.3	60.5	61.8	66.7	20.6	71.8
CAR-MSCR [56]	Cold	0.0	0.0	0.0	33.4	56.5	60.1	62.6	62.6	19.2	72.0
BC-Beta [53]	Cold	0.0	0.0	0.0	50.6	57.2	60.5	62.2	66.7	17.6	71.8
RLSR [36]	Cold	0.0	0.0	0.0	33.3	33.3	61.2	61.9	66.7	21.7	72.7
ORCE-SC [34]	Cold	0.0	0.0	0.0	33.3	33.3	33.3	33.3	66.7	23.6	**75.0**
RLVR [15]	Cold	0.0	0.0	0.0	33.3	33.3	33.3	33.3	33.3	31.0	70.3
RLCR [15]	SFT	0.0	0.0	0.0	33.3	53.0	58.0	82.2	89.9	16.3	72.0
RLCR [15]	Cold	0.0	0.0	0.8	32.9	34.3	57.0	61.4	91.2	19.3	73.1
CARE (Ours)	Cold	**5.4**	**18.2**	**42.4**	**52.9**	**66.8**	**76.4**	**84.9**	**91.4**	**12.5**	73.1

Table 10: HotpotQA-trained models, in-distribution (HotpotQA distractor, 1,000 examples, Llama judge).

Model	C@0	C@1	C@5	C@10	C@15	C@20	C@25	C@30	AURC↓	Acc
RLVR-Classifier [15]	1.6	1.6	14.2	92.8	100.0	100.0	100.0	100.0	6.5	87.7
CAR-WeightedBrier [56]	0.0	0.0	0.0	90.0	**100.0**	**100.0**	**100.0**	**100.0**	6.4	85.8
CAR-MSCR [56]	0.0	0.0	0.0	89.6	**100.0**	**100.0**	**100.0**	**100.0**	7.2	87.0
BC-Beta [53]	0.1	0.1	0.1	89.7	89.7	**100.0**	**100.0**	**100.0**	7.2	81.9
RLSR [36]	0.0	0.0	0.0	85.6	**100.0**	**100.0**	**100.0**	**100.0**	8.6	86.6
ORCE-SC [34]	0.0	0.0	0.0	0.0	**100.0**	**100.0**	**100.0**	**100.0**	10.7	85.6
RLVR [15]	0.0	0.0	0.0	0.0	**100.0**	**100.0**	**100.0**	**100.0**	12.3	**87.7**
RLCR [15]	2.4	2.4	34.7	87.5	**100.0**	**100.0**	**100.0**	**100.0**	6.1	86.1
CARE (Ours)	**6.1**	**6.1**	**74.7**	**93.5**	**100.0**	**100.0**	**100.0**	**100.0**	**4.4**	87.6

answers. For example, consider four completions with confidence scores $(0.9, 0.8, 0.7, 0.6)$. If their correctness labels in this order are (correct, wrong, wrong, correct), accuracy and AUROC are both 50%, and AURC is 41.7%. With labels (wrong, correct, correct, wrong), accuracy and AUROC remain 50%, but AURC increases to 58.3%. Both examples correctly rank two of the four correct–incorrect pairs. However, the second places an incorrect answer first, increasing the risk of accepting only the highest-confidence answers and consequently worsening AURC. Table 12 shows that CARE achieves the highest AUROC in all training and evaluation settings, complementing the coverage results with evidence of better ranking of correct versus incorrect answers.

Calibration. Calibration alone does not ensure useful selective prediction. A model that assigns every answer a confidence equal to its overall accuracy has zero empirical ECE, but cannot distinguish which answers to accept. Even a model that always answers incorrectly and always predicts zero confidence is perfectly calibrated, despite providing no useful answers. Nevertheless, alongside good coverage, calibration can be helpful when abstention thresholds cannot be tuned on labeled validation data and confidence values must be interpreted directly.

Because CARE trains confidence for coverage rather than probability calibration, its confidence scale need not match probabilities of correctness. A simple post-hoc calibration function can adjust this scale. Following Platt-style scaling [44, 24], we fit

$$c_i' = \sigma(a \ \mathrm{logit}(c_i) + b)\,, \qquad a > 0. \tag{10}$$

The positive slope preserves confidence ordering, allowing probability calibration without changing coverage at any risk level or AURC. We fit the two parameters by minimizing binary negative log-likelihood on 1,000 Math training examples or 500 HotpotQA training examples. Before applying the logit, we clip confidence scores to $[10^{-6}, 1 - 10^{-6}]$. This is necessary because the logit is infinite at $c = 0$ and $c = 1$. Clipping moves these endpoint scores slightly inside the interval so that the calibration function can be fitted and evaluated. The fitted function is applied unchanged to ID and OOD evaluation examples.

Table 11: **HotpotQA-trained, out-of-distribution average** over MATH-500, GSM8K, CommonsenseQA, GPQA, SimpleQA, and TriviaQA.

Model	C@0	C@1	C@5	C@10	C@15	C@20	C@25	C@30	AURC↓	Acc
RLVR-Classifier [15]	0.4	0.4	6.0	17.5	17.9	33.7	34.7	36.9	42.0	53.9
CAR-WeightedBrier [56]	0.0	0.0	16.5	32.4	33.4	**44.0**	46.0	46.0	35.6	57.3
CAR-MSCR [56]	0.0	**7.4**	14.8	31.4	32.2	42.7	44.3	44.3	36.2	56.0
BC-Beta [53]	0.9	5.6	14.8	**34.0**	40.7	42.5	**50.0**	50.0	**30.5**	**63.4**
RLSR [36]	0.0	0.0	14.5	32.8	33.3	33.3	33.3	45.6	37.7	58.8
ORCE-SC [34]	0.0	0.0	14.9	16.7	31.7	32.5	43.6	45.4	38.5	55.2
RLVR [15]	0.0	0.0	0.0	16.7	16.7	33.3	33.3	33.3	46.0	53.9
RLCR [15]	0.1	0.1	12.3	26.9	33.7	38.3	40.1	44.5	36.6	56.2
CARE (Ours)	**3.1**	6.4	**23.6**	33.1	**40.8**	42.2	44.6	**50.7**	33.2	58.0

Table 12: **CARE provides the best overall confidence ranking and calibration.** Only in the ID scenario does RLCR achieve a better ECE than CARE, which does not generalize to OOD. RLVR-Classifier is the only method not using verbalized confidence prediction from an LLM. **Bold** and underlining mark the best and second-best results among methods with verbalized confidence within each training setting, including ties.

		ID			OOD		
Training	Method	AUROC↑	Brier↓	ECE↓	AUROC↑	Brier↓	ECE↓
Math	RLVR-Classifier [15]	78.2	.149	9.3	54.6	.343	34.3
	RLVR [15]	46.5	.209	12.3	51.6	.283	25.7
	RLCR [15]	67.8	.166	**8.7**	61.5	.278	25.7
	CARE	**83.7**	**.148**	12.5	**67.3**	**.233**	**22.1**
HotpotQA	RLVR-Classifier [15]	72.1	.102	5.8	58.3	.320	30.1
	RLVR [15]	50.0	.125	13.1	50.1	.291	27.9
	RLCR [15]	78.8	**.098**	**1.9**	67.7	.273	**27.4**
	CARE	**80.7**	**.098**	8.0	**71.4**	**.269**	**27.4**

We evaluate the resulting probabilities using ECE [24] and the binary Brier score [6]:

$$\mathrm{ECE} = \sum_{m=1}^{20} \frac{|B_m|}{N} \left| \frac{1}{|B_m|} \sum_{i \in B_m} y_i - \frac{1}{|B_m|} \sum_{i \in B_m} c'_i \right|, \tag{11}$$

$$\mathrm{Brier} = \frac{1}{N} \sum_{i=1}^{N} (c'_i - y_i)^2, \tag{12}$$

where B_m contains examples whose recalibrated confidence falls in the m-th equal-width bin, with empty bins omitted. ECE measures agreement between average confidence and accuracy within bins, while Brier measures squared probability error. Lower values are better for both. Table 12 shows that CARE achieves the lowest or tied-lowest Brier score after recalibration in all training and evaluation settings. RLCR achieves lower ECE on ID evaluation, but this advantage does not transfer to OOD, where CARE improves or matches its ECE.

A.4 Analysis of confidence distributions of other baselines

Figure 5 extends the Big-Math-Digits confidence-distribution analysis of Figure 4 to the remaining pointwise confidence-learning baselines, CAR-WeightedBrier, CAR-MSCR, and BC-Beta. All three follow the same overall trend as RLCR: mass is heavily concentrated in the high-confidence regime, so these methods remain overconfident in aggregate. They differ from RLCR in two minor ways that the figure makes visible. First, the CAR variants place a small but non-trivial amount of mass near zero, and BC-Beta places a smaller bump there as well, whereas RLCR almost never predicts very low confidence. Second, beyond the single-decimal values that dominate the verbalized baselines,

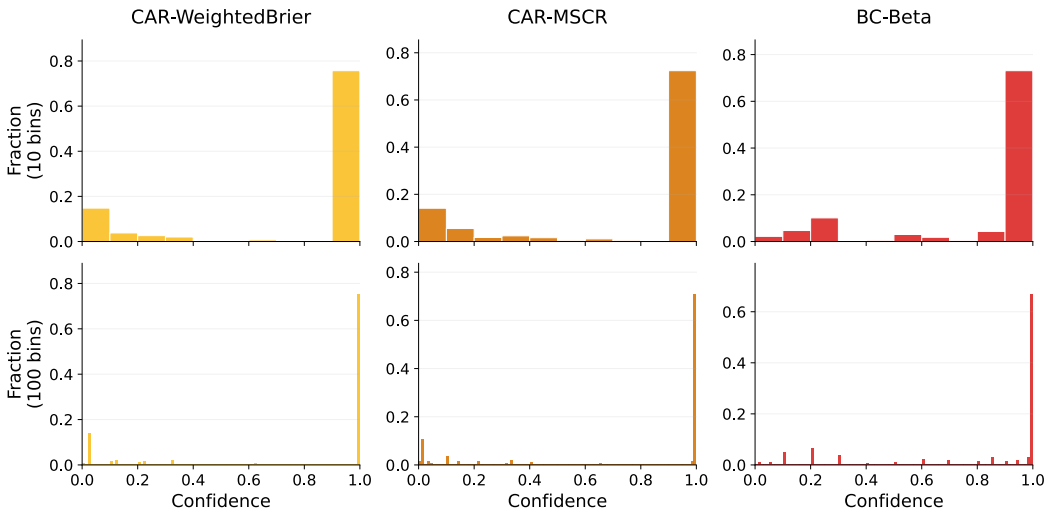


Figure 5: **Big-Math-Digits confidence distributions for the remaining baselines** (CAR-WeightedBrier, CAR-MSCR, BC-Beta), extending Figure 4. Mass concentrates on a small number of bins, in contrast to CARE's finer-grained range.

Table 13: **Reward-geometry ablation:** the exponent controls how strongly larger confidence distances are weighted relative to smaller ones. ID averages Big-Math-Digits, MATH-500, and GSM8K. OOD averages CommonsenseQA, GPQA, SimpleQA, TriviaQA, and HotpotQA.

	ID					OOD				
Reward transform	**C@0**	**C@1**	**C@5**	**AURC↓**	**Acc**	**C@0**	**C@1**	**C@5**	**AURC↓**	**Acc**
Linear (p=1.0)	0.6	0.6	36.1	13.7	72.7	1.2	4.3	**17.1**	**34.9**	**58.8**
Decay exponent (p=0.8)	3.4	9.7	35.3	13.5	72.3	0.4	0.4	16.3	35.8	56.3
Squared (p=2.0)	**5.4**	**18.2**	**42.4**	**12.5**	**73.1**	**2.9**	**5.4**	15.9	37.7	53.6

these methods occasionally use additional values such as 0.95 or 0.85, slightly enlarging the effective vocabulary of confidence scores. The set of distinct values used remains very small compared to CARE, however, so the fine-grained histogram is still dominated by a handful of round-number bins rather than the broader continuous range produced by our method.

A.5 Additional ablations

Table 13 compares exponents applied to the confidence distance from the abstention threshold. Squaring this distance improves ID coverage and AURC, while the linear reward performs better on OOD evaluation.

A.6 Brier reward and training progression

During the first 50 Math training steps, the average absolute advantage contribution before GRPO reward scaling is 0.023 from the Brier reward and 0.011 from the combined coverage rewards, more than twice as much. The ratio falls to 1.08 over steps 51–100, 1.01 over 101–150, and 0.96 over 151–208.

The larger initial contribution does not necessarily mean that CARE's coverage rewards would fail without the Brier term. In fact, on Math, the no-Brier ablation improves ID AURC from 12.5% to 12.1% and OOD AURC from 37.7% to 36.9%, as shown in Table 4. ID accuracy also improves from 73.1% to 73.9%, and OOD accuracy from 53.6% to 56.5%.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the paper's core contribution, the cross-question reward mechanism, and the main empirical improvements. The results and conclusion also state that, although large improvements are observed with the proposed method, gains are not universal and difficult low-risk and OOD settings remain unsolved.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The results discuss rare cases where alternative methods that focus more directly on preserving or improving accuracy can outperform CARE, especially on some OOD settings.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: The paper proposes and evaluates a training reward and does not claim formal theoretical results requiring proofs.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper specifies the datasets, evaluation benchmarks, scoring pipelines, reward design, selected hyperparameters, effective batch geometry, public baseline sources, and the additional ablations needed to reproduce the main claims.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in

some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: The datasets and baseline code used in the work are public, and our implementation is based heavily on the public RLCR codebase. Necessary details to implement our method within the public code repository are presented in the paper, but our version of the code will only be made public at a later date.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The main text and appendix specify the model family, datasets, prompt format, optimizer family, learning rates, schedules, warmup, batch sizes, context lengths, generation counts, selected reward weights, evaluation procedure, and the ablations used to validate the design.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

Justification: The paper evaluates consistently across two distinct training scenarios and multiple in-distribution and out-of-distribution benchmarks, and all reported evaluations use deterministic temperature-0 decoding to reduce sampling noise. Together, this gives a robust picture of the empirical effects even without separate confidence intervals or significance tests.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The appendix states that training was run on NVIDIA H100 and A100 80GB GPUs and that the reported training runs completed in under 24 hours.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work studies confidence estimation and selective prediction using existing benchmark data and models. The draft does not describe data collection from human subjects or other procedures that would require a deviation from the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.

- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The introduction motivates positive uses in assistive and automation settings, discussing both the benefits of better abstention and the risks of over-trust and unsafe over-automation under distribution shift.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not release a new high-risk pretrained model, image generator, or scraped dataset.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [No]

Justification: The draft credits the relevant prior work, datasets, and model families, but it does not yet explicitly enumerate the licenses or terms of use for each external dataset, model, and code asset used.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [N/A]

Justification: The paper does not introduce a new dataset, benchmark, model checkpoint, or code package as a new asset.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The experiments use existing benchmarks and automated or LLM-based evaluation pipelines; the draft does not report new crowdsourcing or human-subjects studies.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.

- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The draft does not involve new human-subjects research, so IRB or equivalent review is not applicable.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.