Unsupervised Confidence Calibration for Reasoning LLMs from a Single Generation

Thomas Zollo [1] **Jimmy Wang** [1] **Richard Zemel** [1]

Abstract

Reasoning language models can solve increasingly complex tasks, but struggle to produce the calibrated confidence estimates necessary for reliable deployment. Existing calibration methods usually depend on labels or repeated sampling at inference time, making them impractical in many settings. We introduce a method for unsupervised confidence calibration of reasoning LLMs when only a single generation is available at inference time. Our approach uses offline sampling on unlabeled data to derive a self-consistency-based proxy target, then distills this signal into a lightweight deployment-time confidence predictor. In a broad evaluation across 5 math and question-answering tasks using 9 reasoning models, our method substantially outperforms baselines, including under distribution shift, and improves downstream performance in selective prediction and simulated downstream decision-making.

1. Introduction

Large language models (LLMs) with reasoning abilities have achieved high accuracy in tasks such as mathematical problem solving and scientific question answering. In consequential settings, however, reliable deployment also requires confidence estimates that reflect the probability that a model's answer is correct (Bommasani et al., 2022; Yang et al., 2026). This property, known as *confidence calibration* (Guo et al., 2017), is widely regarded as a key ingredient for making deep learning systems reliable in practice. Despite its importance, modern LLMs often remain poorly calibrated, making reliable confidence estimation an important unresolved challenge (Kadavath et al., 2022; Leng et al., 2024; Damani et al., 2025).

Although progress has been made, calibration for reasoning LLMs remains difficult to obtain in many of the settings where it would be most useful. Most existing methods, whether based on post-hoc recalibration or RL-style training, require labeled data (Zadrozny & Elkan, 2001; Guo et al., 2017; Kumar et al., 2020; Band et al., 2024; Leng et al., 2024; Damani et al., 2025), while many uncertainty estimation methods rely on repeated generations or other expensive inference-time procedures (Chen et al., 2024; Kuhn et al., 2023; Duan et al., 2024). In practice, however, labeled calibration data is often unavailable or too expensive to obtain (Liu et al., 2022). Repeated-sampling-based confidence estimation is also often impractical: edge devices face strict resource constraints (Wang et al., 2025a), while cloud-served systems may be unable to incur the cost or delay of generating many samples per query.

These limitations arise naturally in real applications. Consider a multilingual math tutor running on students' phones (Figure 1): the system needs calibrated confidence to decide whether to answer directly, offer a hint, or defer to a teacher. In such a setting, labeled data may be difficult to obtain, since collecting correctness labels across many languages, curricula, and student populations would require substantial expert annotation. At the same time, deployment on consumer devices imposes memory, latency, and battery usage constraints, limiting both parallel decoding and the ability to generate multiple samples sequentially. Similar challenges arise in mobile health triage tools, workplace assistants, or embedded accessibility systems, where trustworthy confidence estimates matter but labeled data collection and expensive test-time computation are often out of reach. Further, as LLMs are increasingly deployed as components of agentic systems, lightweight confidence estimates will also be crucial for deciding when to act, defer, invoke tools, or escalate.

In this work, we propose an approach to confidence calibration for reasoning LLMs in a challenging, underserved deployment regime: no labeled calibration data available and only one generation possible. Our key idea is to treat repeated sampling not as a deployment-time inference procedure, but as an offline source of unlabeled supervision.

[1]Department of Computer Science, Columbia University, New York. Correspondence to: Thomas Zollo <tpz2105@columbia.edu>.

Accepted to the ICML 2026 Workshop on Statistical Frameworks for Uncertainty in Agentic Systems, Seoul, South Korea, 2026.
Copyright 2026 by the author(s).

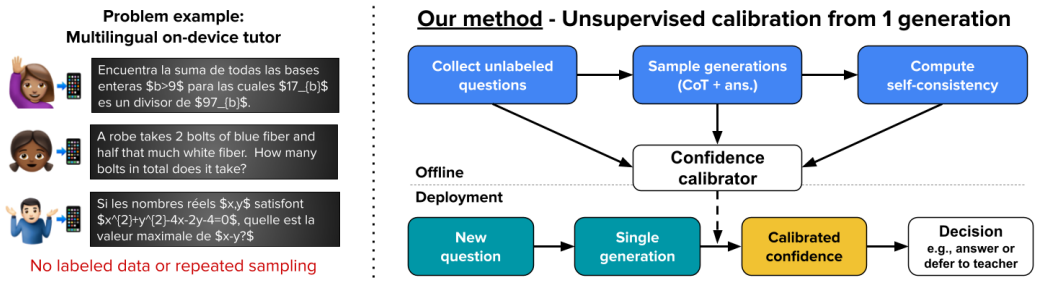

Figure 1. We study calibration for reasoning LLMs in a practical deployment regime where labeled calibration data is unavailable and repeated sampling at test time is too costly. Our approach uses unlabeled questions offline: we sample multiple responses, compute answer self-consistency, and use this signal to train a confidence calibrator. At deployment, the model produces only a single response for a new question, and the calibrator maps that response to a calibrated confidence estimate that can support downstream decisions.

Concretely, we use repeated sampling on unlabeled examples to estimate a self-consistency signal, then amortize that signal into a lightweight predictor that maps a single generated response to an answer-level confidence estimate. This yields a practical route to calibration when labels are unavailable and multi-sample inference is not possible. Requiring only offline access to model outputs, our method is also applicable in black box settings.

We evaluate this approach in comprehensive experiments across 5 math and question-answering tasks using 9 models from the Qwen, DeepSeek, and Nemotron families. Empirically, we show three main results: **(1)** Self-consistency is generally well-calibrated with respect to correctness in reasoning LLMs. **(2)** This signal can be distilled into a single-generation confidence predictor that substantially outperforms relevant unsupervised baselines, including under distribution shift and in black box settings. **(3)** The resulting confidence estimates improve downstream outcomes in applications such as selective prediction (without producing any tokens) and enhancing the calibration of stronger models (e.g., GPT-4o-mini). Our method often narrows much of the gap to stronger supervised or multi-sample approaches, and also works robustly across different temperatures and with a relatively small number of samples (e.g., 5-10).

2. Related Work

Most prior work on calibration in deep learning has focused on classification (Naeini et al., 2015; Guo et al., 2017; Ovadia et al., 2019; Kumar et al., 2020; Minderer et al., 2021; Wang et al., 2021; Zollo et al., 2024), and many of these ideas extend naturally to LLM settings with single-token outputs from a finite discrete set (Desai & Durrett, 2020; Xiao et al., 2022; Yang et al., 2026). Calibration in LLMs becomes substantially more difficult when moving from classification to open-ended generation, where outputs are multi-token, often admit many valid surface forms, and may be supported by reasoning paths. As a result, work in this area has explored a range of answer-level confidence signals for LLMs, including token-probability-based scores, self-evaluation, self-consistency, and direct verbalized confidence estimates (Kadavath et al., 2022; Si et al., 2023; Tian et al., 2023; Xiong et al., 2024; Ulmer et al., 2024; Liu et al., 2024; Lyu et al., 2025). Recent work on reasoning models suggests that improved reasoning performance does not necessarily imply improved calibration, and may even exacerbate overconfidence and poor abstention behavior under some training regimes. This has motivated growing interest in methods that explicitly target confidence estimation in reasoning LLMs, including introspective prompting, verbalized uncertainty, and calibration-aware training objectives (Kirichenko et al., 2025; Mei et al., 2025; Damani et al., 2025; Yoon et al., 2025; Bani-Harouni et al., 2026; Band et al., 2024). Most similar to our approach are recent works that use unlabeled signals from paired base models for calibration (Luo et al., 2025; Tan et al., 2026). Still, these methods are not designed for reasoning models, where long chain-of-thought outputs and greatly enhanced capabilities create a substantial distribution shift for the paired base model; we find that such confidence scores fail consistently across all Qwen3 models and datasets (see Appendix B).

3. Method

We study the problem of assigning calibrated confidence scores to the outputs of reasoning language models when labeled calibration data is unavailable and repeated sampling cannot be used at deployment time. Let x denote an input prompt, and let $y = (r, a) \sim p_\theta(\cdot \mid x)$ denote a full model response, where r is the reasoning trace and a is the extracted final answer. Let $Z \in \{0, 1\}$ indicate whether a is correct. After observing a single response y, our goal is to output a confidence score $c(x, y) \in [0, 1]$ that is calibrated with respect to correctness, i.e. $\mathbb{P}(Z = 1 \mid C = q) = q, \quad \forall q \in [0, 1]$, where C is the random variable corresponding to the reported confidence. In the unsupervised setting, we assume access only to an unlabeled calibration set $\mathcal{D}_{\text{cal}} = \{x_i\}_{i=1}^n$.

Self-Consistency as Confidence Proxy Because labels are unavailable, we cannot directly learn $\mathbb{P}(Z = 1 \mid x, y)$. We therefore seek a proxy for answer confidence. Our key idea is to use *self-consistency* (Wang et al., 2023): if repeated samples from the model tend to yield the same answer, then its answer distribution is concentrated, and confidence is high; if they disagree, confidence is low. Formally, for a fixed input x, let $y^{(1)}, \ldots, y^{(k)} \sim p_\theta(\cdot \mid x)$ be k sampled responses, with each $y^{(j)} = (r^{(j)}, a^{(j)})$. We define the self-consistency score for an input–answer pair as $s(x, a) = \frac{1}{k} \sum_{j=1}^{k} \mathbf{1}\left[a^{(j)} = a\right]$. This is the empirical mass assigned to answer a under the sampled answer distribution. Though not a grounded correctness probability, $s(x, a)$ provides an unlabeled signal of how strongly the model favors that answer, and has been shown to be empirically well-calibrated (Lyu et al., 2025).

Learning a Single-Pass Confidence Predictor Directly computing $s(x, a)$ requires repeated sampling for every query. We instead use self-consistency only offline, on unlabeled data, to construct supervision for a single-pass predictor. For each calibration input $x_i \in \mathcal{D}_{\text{cal}}$, we sample k responses, and use the resulting empirical answer distribution to define proxy confidence scores. We can then select a sampled response $\tilde{y}_i = (\tilde{r}_i, \tilde{a}_i)$ together with its score $s_i = s(x_i, \tilde{a}_i)$, yielding weakly labeled examples of the form $(x_i, \tilde{y}_i, s_i)$. Using these examples, we can train a predictor $f_\phi(x, r, a) \in [0, 1]$ to estimate the empirical support for the response's answer under repeated sampling. At test time, given a new input x and response $y = (r, a)$, we report the confidence score $c(x, y) = f_\phi(x, r, a)$.

4. Experiments

We evaluate our proposed approach along four dimensions. **First,** we test whether self-consistency is a useful unlabeled proxy for top-answer correctness in reasoning LLMs. **Second,** we ask whether this proxy can be distilled into an effective single-generation confidence predictor that outperforms unsupervised baselines. **Third,** we examine whether the resulting predictor remains reliable under distribution shift between unsupervised training and deployment. **Finally,** we assess whether the confidence estimates are useful for downstream decision-making. Below we detail our experimental setup; see Appendix A for additional details.

Reasoning Language Models Our reasoning models include 5 Qwen3 models (0.6B, 1.7B, 4B-Thinking-2507, 8B, 14B) (Yang et al., 2025), 2 Nemotron models (OpenReasoning-Nemotron-7B, Nemotron-Cascade-8B-Thinking) (NVIDIA et al., 2025; Moshkov et al., 2025), and 2 DeepSeek models (DeepSeek-R1-Distill-Llama-8B, DeepSeek-R1-Distill-Qwen-7B) (Guo et al., 2025).

Datasets We focus on datasets with free-form answers (as opposed to multiple choice), which we argue is a natural fit for the difficult unsupervised setting that we are targeting. Our tasks are split between math problem solving and various question answering (QA) tasks, following previous work in LLM calibration (e.g., Damani et al. (2025)). For math, we use **GSM8K** (Cobbe et al., 2021) and **PolyMath** (Wang et al., 2025b). PolyMath is a multilingual dataset, and we use data from 8 languages: EN, ES, FR, PT, IT, RU, AR, ID. We perform science QA with **SciQ** (Welbl et al., 2017), and open-domain QA with **TriviaQA** (Joshi et al., 2017) and **WebQuestions (WebQ)** (Berant et al., 2013). All models are evaluated on the math datasets. For free-form QA, we include only the Qwen3 models and the Nemotron-Cascade-8B-Thinking model (which is derived from Qwen3). The other 3 models failed to reliably follow the formatting required for automatic extraction for these tasks. In total, we run 9 models on 2 math datasets and 6 models on 3 QA datasets for a total of 36 combinations (see Table 6 for model/dataset accuracy). For our experiments we use 1000 examples from each dataset.

Our Method For each data example we generate $k = 100$ responses (reasoning plus answer) with a temperature of 0.7, TopP=0.95, and TopK=20. To build our unsupervised calibration training set we select a response corresponding to the most common answer for each example. The confidence predictor is a weakly supervised version of the typical isotonic regression (Zadrozny & Elkan, 2002) approach to recalibration. We split our calibration set 50/50 to first train an L2 regularized linear model and then train an isotonic regressor from those outputs to the true self-consistency scores (see Figure 1 for sample code). The inputs to the model are embeddings taken from the reasoning model itself after producing the last token of a new response (i.e., based on the entire input context). This enables seamless and lightweight integration with existing generation pipelines.

Evaluation We measure calibration using expected calibration error (both ECE1 and ECE2) with 12 equal-mass bins, as well as maximum calibration error (MCE), which measures the worst calibration over any bin (Guo et al., 2017; Kumar et al., 2020). We also measure Brier score (BS) (Brier, 1950) and AUROC; while AUROC does not measure calibration, it captures another important quality for useful confidence estimates, that they effectively separate correct and incorrect answers (Kadavath et al., 2022).

Baselines The most common baseline for producing LLM confidence estimates is the length-normalized token probability for the full model response (e.g., the geometric mean of the top token probabilities) (Kadavath et al., 2022; Luo et al., 2025). For reasoning models, to avoid scores being distorted by long thinking chains with many unimportant

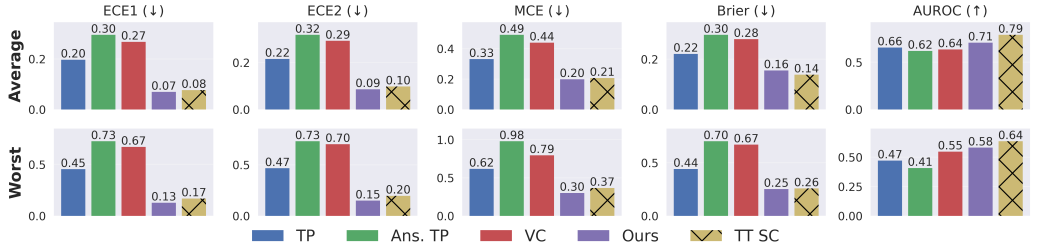

Figure 2. Average (top) and worst-case (bottom) calibration results for our method and baselines across 36 model/dataset combinations. Besides baselines for our unsupervised setting (Token Probs., Answer Probs., Verbal Conf.), we also measure the effectiveness of test-time self-consistency. Our method significantly outperforms baselines and offers competitive performance to the more expensive methods.

tokens, one might report the length-normalized token probability of only the tokens associated with the final answer (Damani et al., 2025). Given that both of these baselines are viable without labels or multiple samples, we adopt them (**Token Probs. (TP)** and **Answer Probs. (Ans. TP)**) for comparison in our setting. Another popular approach to LLM calibration involves having the model directly verbalize its own confidence in its output (**Verbal Conf.** (VC)) (Tian et al., 2023). While this does in fact violate our constraints since it requires another full response (including reasoning), we include it because of its prevalence in the LLM calibration literature. Further, to ensure the strongest verbalized confidence baseline possible, we search for our prompt among 5 prompts written by GPT-5.2 and 5 written by Gemini-3, and select the prompt with the best results across a selection of models and datasets (see Appendix A.3.2 for more details and the final prompt). Finally, we also include results for stronger baselines that violate the constraints of our setting. Namely, we include (1) a **Supervised (Sup.)** baseline where we perform recalibration via Platt scaling (Platt, 1999) on the token probabilities and (2) a multiple-sampling baseline measuring true **Test-Time Self-Consistency (TT SC)** with 100 samples.

4.1. Main Results

In our first experiment we study our first two questions: (1) does answer self-consistency in reasoning LLMs generally provide a well-calibrated confidence estimate? (2) can this signal be learned by a lightweight test-time predictor? We focus on the setting where we have unlabeled calibration data from the same distribution as the test data, as is typical in unsupervised learning (Ganin et al., 2016; Luo et al., 2025). For each model and dataset, we average calibration results over 200 random trials. In each trial, we randomly split 1000 examples 40/60 between calibration and test sets. We use the unlabeled calibration data to train our calibration model, and measure results on the test set. Figure 2 shows average and worst-case results for all methods over all model/dataset combinations. Table 1 shows results grouped by model family and task type. We report more granular results in the Appendix: Table 7 reports results by dataset, and Tables 8 and 9 report results by model.

First, we focus on question (1), whether self-consistency is in fact a good signal for calibrated confidence in reasoning LLMs. In both Figure 2 and Table 1, test-time self-consistency with 100 samples achieves strong calibration across model families and task types, with low ECE and Brier score on average and reasonable worst-case scores. In particular, its performance is competitive with or better than the supervised baseline, suggesting that agreement across sampled responses captures a substantial amount of information about correctness. Further, we see that self-consistency is always better than Platt scaling the token probabilities with respect to Brier score and AUROC (see also Table 7), meaning it generally produces sharper and more discriminative confidence estimates that can be useful for decision-making. Taken together, these results support the central premise of our method: self-consistency is a high-quality target for learning answer-level confidence in reasoning LLMs.

Second, we examine whether our lightweight calibrator improves on existing baselines. Figure 2 shows that our method is the strongest approach among the methods that satisfy our deployment constraints, both on average and in the worst case. In fact, the worst ECE1, ECE2, or MCE incurred by our method across 36 model/dataset combinations is better than the average achieved by any baseline (and nearly so for Brier score). Table 1 shows the same pattern across model families and task types: relative to token probabilities, answer-token probabilities, and verbalized confidence, our method substantially reduces ECE2 and Brier score in every group. Notably, our method often approaches the performance of test-time self-consistency, despite requiring only a single generation at inference time, and in some groups it is competitive with the supervised baseline. Appendix Tables 7, 8, and 9 show that this pattern is not driven by a small subset of settings, but instead holds broadly across individual datasets and models.

Qualitative Results Figure 3 shows reliability diagrams with confidence histograms for DeepSeek-R1-Distill-Llama-

	Qwen		Nemotron		DeepSeek		Math		QA	
Method	ECE2	BS	ECE2	BS	ECE2	BS	ECE2	BS	ECE2	BS
TP	0.217	0.223	0.191	0.210	0.250	0.234	0.116	0.128	0.315	0.315
Ans. TP	0.333	0.316	0.256	0.250	0.336	0.259	0.182	0.147	0.455	0.447
VC	0.304	0.296	0.284	0.255	0.243	0.222	0.165	0.150	0.421	0.410
Ours	**0.089**	**0.161**	**0.083**	**0.142**	**0.076**	**0.140**	**0.075**	**0.102**	**0.097**	**0.208**
Black Box	0.090	0.160	0.083	0.145	0.077	0.139	0.077	0.100	0.097	0.210
TT SC	0.103	0.145	0.096	0.130	0.067	0.116	0.078	0.083	0.118	0.194
Sup.	0.063	0.161	0.063	0.152	0.083	0.170	0.058	0.110	0.073	0.211

Table 1. Results broken down by different groupings of models (Qwen, Nemotron, DeepSeek) and datasets (math and QA), including baselines for our unsupervised setting (Token Probs., Answer Probs., Verbal Conf.), results from a black box setting where the input embeddings come from an external model, and strong (supervised and test-time self-consistency) baselines that do not meet our constraints of no labels and one generation. Our method is effective across splits, including in the black box setting.

8B. Reliability diagrams visualize calibration by plotting confidence vs. accuracy for each bin of data; a well-calibrated model produces a curve that lies on or close to the line $x = y$, where confidence matches accuracy. We augment these with histograms to show the distribution of confidence estimates by each method. The plots show that token-probability baselines typically exhibit compressed confidence ranges that are untethered from the actual model accuracy, while verbalized confidence estimates also fail to reflect true correctness. Answer tokens cluster towards having extremely high probability, likely because the reasoning chains already contain the eventual answer before these final tokens are decoded. In contrast, our method produces curves that more closely track the diagonal, and allocates mass across confidence levels in a way that reflects actual correctness probabilities. These plots are consistent with the reductions in ECE and Brier score, and suggest our model is capturing meaningful variation in example difficulty. We provide plots for 3 more models in Figures 8, 9, and 10, which support our findings here.

Calibration Error and Accuracy Previous work (Xiong et al., 2024), as well as our experiments (e.g., top row of Figure 3), has shown that existing baselines can appear reasonably calibrated according to ECE simply because they are almost always highly confident and the underlying models are often fairly accurate on the tasks under consideration. For example, a method that mostly outputs confidences near 0.85 can achieve at least modest ECE on any task with accuracy in the range of, e.g., 70-100%. To test whether our method is benefiting from a similar effect, we examine how calibration changes as the underlying model accuracy varies across the five Qwen3 model sizes. In Figure 4, we plot accuracy against ECE2 for each method and dataset across the Qwen3 model family. We see that the baselines are often highly sensitive to changes in model accuracy: as accuracy decreases, their calibration error increases substantially, consistent with these methods relying on poorly adapted or weakly informative confidence scales. By contrast, our method remains much more stable and tends to achieve low ECE2 across a broader range of accuracies. This suggests that its performance is not simply an artifact of incidentally approximating the average accuracy level of the task, but instead reflects a more faithful estimate of correctness probability.

Black Box Setting In our main implementation, our model takes as input the embeddings from the LLM producing the response, highlighting easy integration into existing pipelines. However, the algorithm in Section 3 does not require such "white-box" access to the reasoning model. It only requires the ability to sample outputs, making it applicable in "black box" settings where model internals are unavailable. We therefore next test whether the method remains effective in this setting, or whether our earlier results depend on access to the answering model's embeddings. To do so, we use embeddings from an external model, Gemma3-4B-it, while otherwise running the method exactly as before for all model/dataset combinations. Results in Table 1 show that this variant yields nearly identical average performance. This suggests that access to the reasoning model's own hidden states is not essential for strong performance; rather, we only need access to a reasonably strong embedding model.

Hyperparameter Ablations Our method has two main hyperparameters: the temperature used for offline sampling and the number of samples used to construct the targets. In our experiments, we use a temperature of 0.7, which lies in the typical range for sampling from reasoning LLMs (Yang et al., 2025; Guo et al., 2025). For the number of samples, we set $k = 100$, a value large enough to study the performance of the underlying method without introducing additional variance from small answer sets. To assess robustness to these design choices, we perform ablations over both hyperparameters; results are presented in Appendix C.1.1. **We find that our method substantially outperforms the**

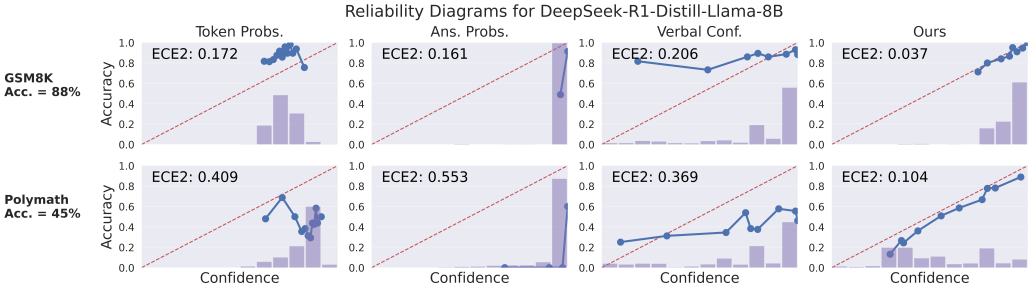

Figure 3. Reliability diagrams are used to visualize calibration. For each method, the blue line shows binned confidence vs. accuracy; perfect calibration lies on the red $x = y$ line. Purple histogram bars plot the distribution of scores produced by the method. Our approach produces confidence estimates that are calibrated with respect to actual correctness.

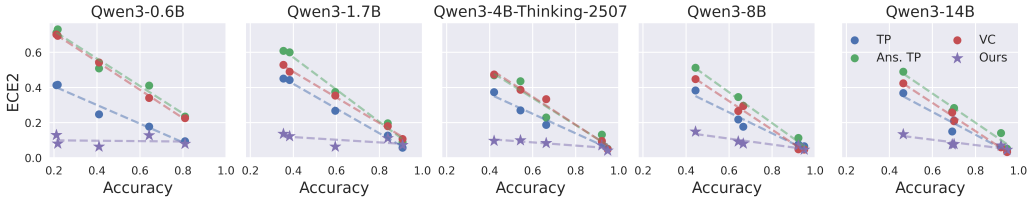

Figure 4. Examining the relationship between accuracy and calibration error for various methods. A flat slope indicates that a method remains effective even when accuracy is low. Existing methods are sensitive to accuracy, yielding large negative slopes where calibration error increases quickly on more difficult tasks; our method sees only slight degradation.

best unsupervised baseline with as few as 5 samples (reducing ECE by over 50%), continues to improve up to our largest setting of $k = 100$, and remains robust to temperature choices in {0.5, 0.7, 0.9}.

4.2. Calibration Under Distribution Shift

Next, we study whether our method is reliable when faced with shifts in the distribution between calibration and test data. To do so, we evaluate two types of distribution shifts. First, using the PolyMath dataset, we study distribution shifts across groupings of the 8 languages in our data sample. We create 4 different splits based on various language groupings (for example, romance vs. non-romance), and trial each split as both training and test data, for a total of 8 experiments (see Appendix C.2 for more details on the language splits). Second, we study distribution shifts across task domains. For math, we study transfer between PolyMath and GSM8K (2 combinations), and for QA we study transfer among SciQ, TriviaQA, and WebQ (6 combinations). We perform the experiments with all models with 4B parameters or more, where such robustness might be expected. Results are summarized in Table 2, and presented in detail in Appendix C.2.

Our method remains robust under all three forms of distribution shift, and is consistently the strongest unsupervised approach. Under language shift in PolyMath, it achieves the best calibration with an ECE2/Brier score of 0.09/0.12, improving substantially over token probabilities (0.15/0.17), answer-token probabilities (0.24/0.19), and verbalized confidence (0.18/0.18). We see a similar pattern under cross-dataset transfer in math, where our method again performs best. In the challenging QA setting our approach yields the clearest advantage, reducing ECE2 to 0.13 versus 0.28 for the best baseline, while also achieving the lowest Brier score. Taken together, these results suggest that the learned predictor is not merely fitting a narrow in-domain confidence scale, but instead captures signals that transfer across both linguistic and task-level shifts.

4.3. Usefulness for Decision-Makers

Sections 4.1 and 4.2 show that our method produces substantially better-calibrated confidence estimates than existing unsupervised baselines both in-domain and under distribution shift. However, calibration is most valuable when it supports better downstream decisions and outcomes (Bo et al., 2025). We therefore next study two complementary decision-making settings: selective prediction (Geifman & El-Yaniv, 2017), where confidence is used to abstain on uncertain examples, and linguistic calibration (Band et al., 2024), where confidence statements help a downstream decision-maker form better-calibrated judgments. Together, these experiments test whether our confidence estimates are not only calibrated in isolation, but also actionable for systems that must decide when to trust, defer, or hedge.

Selective Prediction Selective prediction uses confidence estimates to decide which examples to answer and which to abstain on, with the goal of improving accuracy on the

	Language		Math Domain		QA Domain	
Method	ECE2	BS	ECE2	BS	ECE2	BS
Token Probs.	0.153	0.165	0.113	0.121	0.283	0.301
Ans. Probs.	0.238	0.190	0.171	0.133	0.386	0.384
Verbal Conf.	0.182	0.181	0.151	0.138	0.353	0.345
Ours	**0.092**	**0.123**	**0.095**	**0.111**	**0.134**	**0.232**

Table 2. Calibration results under distribution shift in language (using PolyMath) as well as across domains in math and QA; scores are averaged across models and shifts. Our method offers reliable performance even when calibration data does not match test data.

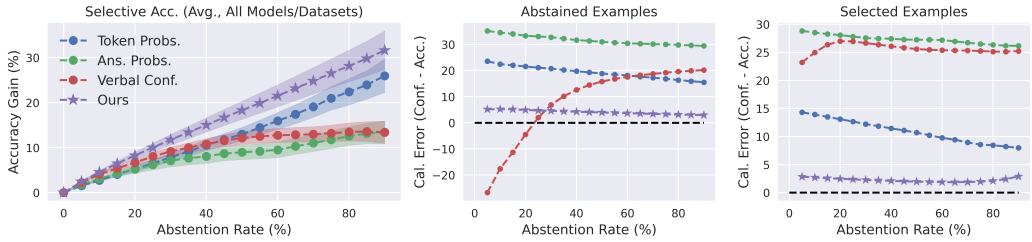

Figure 5. On the left, results for selective prediction using our method and baselines. The X-axis reflects the percent of examples on which the system abstains, and the y-axis reflects the corresponding gain in accuracy. In the middle and right plots, we examine the difference between the average confidence and accuracy of the abstained and selected examples for each method at each level of abstention. Our method brings gains in accuracy while allowing the decision-maker to understand the likely accuracy on rejected and accepted predictions.

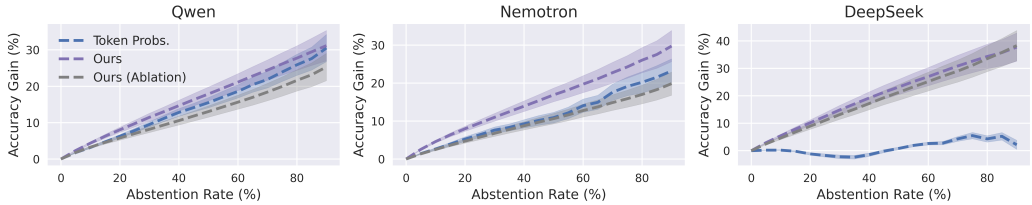

Figure 6. Selective prediction results by model family. Results include an ablation of our method where only the question is used for confidence estimation, allowing for abstention without producing the many tokens associated with reasoning outputs.

selected examples by abstaining on the most uncertain cases (Geifman & El-Yaniv, 2017; 2019; Yadkori et al., 2024). Although this is not strictly dependent on calibration (e.g., selective accuracy is invariant to scaling of the confidence), the ability to order examples from least to most likely to be correct is another desirable quality in a confidence estimator. To study whether our method is useful in this way, we use the same experimental setup as Section 4.1, using the unlabeled calibration data to train our confidence estimation model. Using our method and baselines, we rank test examples by confidence and evaluate abstention rates from 0% to 90%, measuring the resulting gain in accuracy on the non-abstained set relative to the full test set. We run 100 trials for each of the 36 model/dataset combinations, with 1000 examples and a 40/60 split between calibration and test data. Selective accuracy gains are averaged over the 3600 trials, and we also analyze the abstained and selected sets by comparing their average confidence and realized accuracy; this lets us test not only whether a method selects good examples, but also whether it provides a calibrated picture of the quality of both the rejected and accepted examples.

Figures 5 and 6 show that our method yields the strongest overall selective prediction performance among the methods available in our setting. In Figure 5 (left), abstaining according to our confidence estimates produces substantially larger gains in retained-set accuracy across nearly the full range of abstention rates, whereas the baselines improve more slowly and begin to plateau earlier. Moreover, the middle and right plots in Figure 5 show that, for both the abstained and selected sets, our method's average confidence tracks realized accuracy much more closely than the alternatives, enabling a decision-maker to understand the likely accuracy of both rejected and accepted predictions. Figure 6 shows that these gains hold across model families, while the strongest baseline shows unstable performance.

Additionally, we ablate our method so that the confidence predictor is given only the question embedding, rather than features from a generated response. This would allow abstention decisions to be made before producing the (long) reasoning output, potentially complementing gains in accuracy with inference cost savings roughly equal to the abstention rate. Results are shown in Figure 6. The question-only

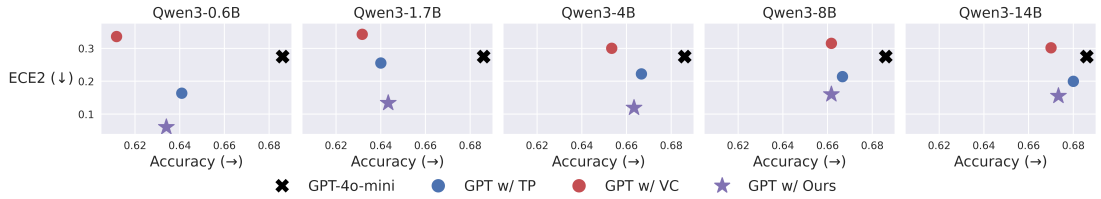

Figure 7. Linguistic calibration results when a stronger decision-maker LLM (GPT-4o-mini) is given answers and confidence estimates from a smaller assistant LLM (Qwen3). Our method consistently produces the most calibrated downstream predictions, showing that its confidence estimates are useful not only in isolation but also when communicated to another (stronger) model.

ablation remains competitive with, and often outperforms, the best unsupervised baseline. This result is especially promising, **highlighting the potential of our approach to deliver significant performance and efficiency gains without any labels.**

Linguistic Calibration Finally, we evaluate our confidence estimates through the lens of downstream decision-making via linguistic calibration (Band et al., 2024). Concretely, we treat the reasoning model as an assistant whose answer and confidence are provided to a separate decision-maker, and examine whether this information enables the decision-maker to produce better-calibrated predictions. We simulate the decision-maker (as in (Band et al., 2024)) with GPT-4o-mini, use SciQ as the evaluation dataset, and repeat the experiment five times, using each Qwen model in turn as the underlying assistant model (all of which are less accurate than GPT-4o-mini on SciQ). We then compare the calibration (and accuracy) of the decision-maker's final predictions when given no assistance, or when conditioned on answers paired with confidence expressions from token probabilities, verbalized confidence, or our method. Figure 7 and Table 14 show that our method consistently leads to the best-calibrated downstream decisions. Averaged over the five Qwen assistants, conditioning GPT-4o-mini on our confidence estimates reduces ECE2 to 0.13 on average, compared to 0.21 for token probabilities, 0.32 for verbalized confidence, and 0.27 for GPT-4o-mini without any confidence input; it also yields the best average Brier score and MCE. This pattern holds across all five models, with our method producing the lowest ECE2 in every case. Overall, using predictions from the less accurate assistants leads to a relatively small drop in accuracy, but meaningful gains in calibration; more accurate assistant models lead to higher prediction accuracy. These results suggest that the benefit of our approach extends to helping more accurate downstream decision-makers form more reliable beliefs about answer correctness.

5. Discussion and Limitations

In our experiments, we find that calibrated confidence for reasoning LLMs need not require either labeled calibration data (and possibly compute-intensive RL training) or expensive test-time sampling. Instead, we can achieve a useful middle ground: a strong but costly uncertainty signal can be computed offline on unlabeled data (for example overnight while a device is plugged into a power source), then distilled into a lightweight predictor that is practical at deployment. Importantly, this advantage is not limited to open-weight models: our black box results suggest that the same strategy can remain effective even when the underlying reasoning model is only accessible through a proprietary interface. Our results also highlight that verbalized confidence remains a weak signal for calibration. Across our experiments, verbalized confidence is consistently outperformed by our method and simple token probabilities, and often performs poorly in absolute terms, in line with prior findings that linguistic expressions of confidence need not correspond to calibrated beliefs (Zhou et al., 2023; Xiong et al., 2024).

A limitation of our method is that it relies on offline sampling, which may be resource-intensive and need to be repeated over time (though Table 10 shows that sampling as few as 5 times leads to significant ECE and Brier score reductions). More fundamentally, the method can only be as good as the self-consistency signal it distills: if agreement among samples is weakly related to correctness, or models are consistently wrong with high internal agreement, the learned predictor may inherit those failures (although one might argue that such a model should not be deployed anyway). Our study also has limitations. While we perform a comprehensive evaluation, the experiments are confined to relatively well-defined settings with automatic evaluation, and do not test more open-ended tasks, interactive multi-turn deployments, or decision problems with richer utilities than binary correctness. Last, our downstream studies should only be viewed as evidence of promise rather than a complete picture of real-world decision-making usefulness.

Reproducibility

All parameters and prompts used to produce our data and results are contained in this section and Appendix A. Our study relies entirely on open source models and datasets, and our code is available at `https://github.com/thomaspzollo/llm_unsupervised_conf`.

References

Band, N., Li, X., Ma, T., and Hashimoto, T. Linguistic calibration of language models, 2024.

Bani-Harouni, D., Pellegrini, C., Stangel, P., Özsoy, E., Zaripova, K., Navab, N., and Keicher, M. Rewarding doubt: A reinforcement learning approach to calibrated confidence expression of large language models, 2026. URL https://arxiv.org/abs/2503.02623.

Berant, J., Chou, A., Frostig, R., and Liang, P. Semantic parsing on Freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1533–1544, Seattle, Washington, USA, October 2013. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D13-1160.

Bo, J. Y., Wan, S., and Anderson, A. To rely or not to rely? evaluating interventions for appropriate reliance on large language models. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pp. 1–23, 2025.

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N., Chen, A., Creel, K., Davis, J. Q., Demszky, D., Donahue, C., Doumbouya, M., Durmus, E., Ermon, S., Etchemendy, J., Ethayarajh, K., Fei-Fei, L., Finn, C., Gale, T., Gillespie, L., Goel, K., Goodman, N., Grossman, S., Guha, N., Hashimoto, T., Henderson, P., Hewitt, J., Ho, D. E., Hong, J., Hsu, K., Huang, J., Icard, T., Jain, S., Jurafsky, D., Kalluri, P., Karamcheti, S., Keeling, G., Khani, F., Khattab, O., Koh, P. W., Krass, M., Krishna, R., Kuditipudi, R., Kumar, A., Ladhak, F., Lee, M., Lee, T., Leskovec, J., Levent, I., Li, X. L., Li, X., Ma, T., Malik, A., Manning, C. D., Mirchandani, S., Mitchell, E., Munyikwa, Z., Nair, S., Narayan, A., Narayanan, D., Newman, B., Nie, A., Niebles, J. C., Nilforoshan, H., Nyarko, J., Ogut, G., Orr, L., Papadimitriou, I., Park, J. S., Piech, C., Portelance, E., Potts, C., Raghunathan, A., Reich, R., Ren, H., Rong, F., Roohani, Y., Ruiz, C., Ryan, J., Ré, C., Sadigh, D., Sagawa, S., Santhanam, K., Shih, A., Srinivasan, K., Tamkin, A., Taori, R., Thomas, A. W., Tramèr, F., Wang, R. E., Wang, W., Wu, B., Wu, J., Wu, Y., Xie, S. M., Yasunaga, M., You, J., Zaharia, M., Zhang, M., Zhang, T., Zhang, X., Zhang, Y., Zheng, L., Zhou, K., and Liang, P. On the opportunities and risks of foundation models, 2022. URL https://arxiv.org/abs/2108.07258.

Brier, G. W. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950.

Chen, C., Liu, K., Chen, Z., Gu, Y., Wu, Y., Tao, M., Fu, Z., and Ye, J. Inside: Llms' internal states retain the power of hallucination detection, 2024.

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., Hesse, C., and Schulman, J. Training verifiers to solve math word problems, 2021. URL https://arxiv.org/abs/2110.14168.

Damani, M., Puri, I., Slocum, S., Shenfeld, I., Choshen, L., Kim, Y., and Andreas, J. Beyond binary rewards: Training lms to reason about their uncertainty. *arXiv preprint arXiv:2507.16806*, 2025.

Desai, S. and Durrett, G. Calibration of pre-trained transformers, 2020.

Duan, J., Cheng, H., Wang, S., Zavalny, A., Wang, C., Xu, R., Kailkhura, B., and Xu, K. Shifting attention to relevance: Towards the predictive uncertainty quantification of free-form large language models, 2024.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., and Lempitsky, V. Domain-adversarial training of neural networks, 2016. URL https://arxiv.org/abs/1505.07818.

Geifman, Y. and El-Yaniv, R. Selective classification for deep neural networks, 2017.

Geifman, Y. and El-Yaniv, R. Selectivenet: A deep neural network with an integrated reject option, 2019.

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. On calibration of modern neural networks, 2017.

Guo, D., Yang, D., Zhang, H., Song, J., Wang, P., Zhu, Q., Xu, R., Zhang, R., Ma, S., Bi, X., Zhang, X., Yu, X., Wu, Y., Wu, Z. F., Gou, Z., Shao, Z., Li, Z., Gao, Z., Liu, A., Xue, B., Wang, B., Wu, B., Feng, B., Lu, C., Zhao, C., Deng, C., Ruan, C., Dai, D., Chen, D., Ji, D., Li, E., Lin, F., Dai, F., Luo, F., Hao, G., Chen, G., Li, G., Zhang, H., Xu, H., Ding, H., Gao, H., Qu, H., Li, H., Guo, J., Li, J., Chen, J., Yuan, J., Tu, J., Qiu, J., Li, J., Cai, J. L., Ni, J., Liang, J., Chen, J., Dong, K., Hu, K., You, K., Gao, K., Guan, K., Huang, K., Yu, K., Wang, L., Zhang, L., Zhao, L., Wang, L., Zhang, L., Xu, L., Xia, L., Zhang, M., Zhang, M., Tang, M., Zhou, M., Li, M., Wang, M., Li, M., Tian, N., Huang, P., Zhang, P., Wang, Q., Chen, Q., Du, Q., Ge, R., Zhang, R., Pan, R., Wang, R., Chen, R. J., Jin, R. L., Chen, R., Lu, S., Zhou, S., Chen, S., Ye, S., Wang, S., Yu, S., Zhou, S., Pan, S., Li, S. S., Zhou, S., Wu, S., Yun, T., Pei, T., Sun, T., Wang, T., Zeng, W., Liu, W., Liang, W., Gao, W., Yu, W., Zhang, W., Xiao, W. L., An, W., Liu, X., Wang, X., Chen, X., Nie, X., Cheng, X., Liu, X., Xie, X., Liu, X., Yang, X., Li, X., Su, X., Lin, X.,

Li, X. Q., Jin, X., Shen, X., Chen, X., Sun, X., Wang, X., Song, X., Zhou, X., Wang, X., Shan, X., Li, Y. K., Wang, Y. Q., Wei, Y. X., Zhang, Y., Xu, Y., Li, Y., Zhao, Y., Sun, Y., Wang, Y., Yu, Y., Zhang, Y., Shi, Y., Xiong, Y., He, Y., Piao, Y., Wang, Y., Tan, Y., Ma, Y., Liu, Y., Guo, Y., Ou, Y., Wang, Y., Gong, Y., Zou, Y., He, Y., Xiong, Y., Luo, Y., You, Y., Liu, Y., Zhou, Y., Zhu, Y. X., Huang, Y., Li, Y., Zheng, Y., Zhu, Y., Ma, Y., Tang, Y., Zha, Y., Yan, Y., Ren, Z. Z., Ren, Z., Sha, Z., Fu, Z., Xu, Z., Xie, Z., Zhang, Z., Hao, Z., Ma, Z., Yan, Z., Wu, Z., Gu, Z., Zhu, Z., Liu, Z., Li, Z., Xie, Z., Song, Z., Pan, Z., Huang, Z., Xu, Z., Zhang, Z., and Zhang, Z. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, September 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09422-z. URL http://dx.doi.org/10.1038/s41586-025-09422-z.

Joshi, M., Choi, E., Weld, D. S., and Zettlemoyer, L. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension, 2017. URL https://arxiv.org/abs/1705.03551.

Kadavath, S., Conerly, T., Askell, A., Henighan, T., Drain, D., Perez, E., Schiefer, N., Hatfield-Dodds, Z., DasSarma, N., Tran-Johnson, E., Johnston, S., El-Showk, S., Jones, A., Elhage, N., Hume, T., Chen, A., Bai, Y., Bowman, S., Fort, S., Ganguli, D., Hernandez, D., Jacobson, J., Kernion, J., Kravec, S., Lovitt, L., Ndousse, K., Olsson, C., Ringer, S., Amodei, D., Brown, T., Clark, J., Joseph, N., Mann, B., McCandlish, S., Olah, C., and Kaplan, J. Language models (mostly) know what they know, 2022.

Kirichenko, P., Ibrahim, M., Chaudhuri, K., and Bell, S. J. Abstentionbench: Reasoning llms fail on unanswerable questions. *arXiv preprint arXiv:2506.09038*, 2025.

Kuhn, L., Gal, Y., and Farquhar, S. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation, 2023.

Kumar, A., Liang, P., and Ma, T. Verified uncertainty calibration, 2020. URL https://arxiv.org/abs/1909.10155.

Kwon, W., Li, Z., Zhuang, S., Sheng, Y., Zheng, L., Yu, C. H., Gonzalez, J. E., Zhang, H., and Stoica, I. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

Leng, J., Huang, C., Zhu, B., and Huang, J. Taming overconfidence in llms: Reward calibration in rlhf. *arXiv preprint arXiv:2410.09724*, 2024.

Liu, X., Yoo, C., Xing, F., Oh, H., Fakhri, G. E., Kang, J.-W., and Woo, J. Deep unsupervised domain adaptation: A review of recent advances and perspectives. *arXiv preprint arXiv:2208.07422*, 2022.

Liu, X., Khalifa, M., and Wang, L. Litcab: Lightweight language model calibration over short- and long-form responses, 2024.

Luo, B., Wang, S., Li, S., and Wei, H. Your pre-trained llm is secretly an unsupervised confidence calibrator, 2025. URL https://arxiv.org/abs/2505.16690.

Lyu, Q., Shridhar, K., Malaviya, C., Zhang, L., Elazar, Y., Tandon, N., Apidianaki, M., Sachan, M., and Callison-Burch, C. Calibrating large language models with sample consistency. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025.

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

Mei, Z., Zhang, C., Yin, T., Lidard, J., Shorinwa, O., and Majumdar, A. Reasoning about uncertainty: Do reasoning models know when they don't know?, 2025. URL https://arxiv.org/abs/2506.18183.

Minderer, M., Djolonga, J., Romijnders, R., Hubis, F., Zhai, X., Houlsby, N., Tran, D., and Lucic, M. Revisiting the calibration of modern neural networks, 2021. URL https://arxiv.org/abs/2106.07998.

Moshkov, I., Hanley, D., Sorokin, I., Toshniwal, S., Henkel, C., Schifferer, B., Du, W., and Gitman, I. Aimo-2 winning solution: Building state-of-the-art mathematical reasoning models with openmathreasoning dataset, 2025. URL https://arxiv.org/abs/2504.16891.

Naeini, M. P., Cooper, G., and Hauskrecht, M. Obtaining well calibrated probabilities using bayesian binning. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

NVIDIA, :, Blakeman, A., Grattafiori, A., Basant, A., Gupta, A., Khattar, A., Renduchintala, A., Vavre, A., Shukla, A., Bercovich, A., Ficek, A., Shaposhnikov, A., Kondratenko, A., Bukharin, A., Milesi, A., Taghibakhshi, A., Liu, A., Barton, A., Mahabaleshwarkar, A. S., Klein, A., Zuker, A., Geifman, A., Shen, A., Bhiwandiwalla, A., Tao, A., Agrusa, A., Verma, A., Guan, A., Mandarwal, A., Mehta, A., Aithal, A., Poojary, A., Ahamed, A., Mishra, A., Thekkumpate, A. K., Dattagupta, A., Zhu, B., Sadeghi, B., Simkin, B., Lanir, B., Schifferer, B., Nushi, B., Kartal, B., Rouhani, B. D., Ginsburg, B., Norick, B., Soubasis, B., Kisacanin, B., Yu, B., Catanzaro, B., del Mundo, C., Hwang, C., Wang, C., Hsieh, C.-P., Zhang, C., Yu, C., Mungekar, C., Patel, C., Alexiuk, C., Parisien, C., Neale, C., Meurillon, C., Mosk-Aoyama, D., Su, D., Corneil, D., Afrimi, D., Lo, D., Rohrer, D., Serebrenik, D., Gitman, D., Levy, D., Stosic, D., Mosallanezhad, D., Narayanan, D., Nathawani, D., Rekesh, D., Yared, D., Kakwani, D.,

Ahn, D., Riach, D., Stosic, D., Minasyan, E., Lin, E., Long, E., Long, E. P., Segal, E., Lantz, E., Evans, E., Ning, E., Chung, E., Harper, E., Tramel, E., Galinkin, E., Pounds, E., Briones, E., Bakhturina, E., Tsykunov, E., Ladhak, F., Wang, F., Jia, F., Soares, F., Chen, F., Galko, F., Sun, F., Siino, F., Agam, G. H., Ajjanagadde, G., Bhatt, G., Prasad, G., Armstrong, G., Shen, G., Batmaz, G., Nalbandyan, G., Qian, H., Sharma, H., Ross, H., Ngo, H., Hum, H., Sahota, H., Wang, H., Soni, H., Upadhyay, H., Mao, H., Nguyen, H. C., Nguyen, H. Q., Cunningham, I., Galil, I., Shahaf, I., Gitman, I., Loshchilov, I., Schen, I., Levy, I., Moshkov, I., Golan, I., Putterman, I., Kautz, J., Scowcroft, J. P., Casper, J., Mitra, J., Glick, J., Chen, J., Oliver, J., Zhang, J., Zeng, J., Lou, J., Zhang, J., Choi, J., Huang, J., Conway, J., Guman, J., Kamalu, J., Greco, J., Cohen, J., Jennings, J., Daw, J., Vialard, J. V., Yi, J., Parmar, J., Xu, K., Zhu, K., Briski, K., Cheung, K., Luna, K., Wyss, K., Santhanam, K., Shih, K., Kong, K., Bhardwaj, K., Shankar, K., Puvvada, K. C., Pawelec, K., Anik, K., McAfee, L., Sleiman, L., Derczynski, L., Ding, L., Wei, L., Liebenwein, L., Vega, L., Grover, M., Segbroeck, M. V., de Melo, M. R., Nazemi, M., Sreedhar, M. N., Kilaru, M., Ashkenazi, M., Romeijn, M., Chochowski, M., Cai, M., Kliegl, M., Moosaei, M., Kulka, M., Novikov, M., Samadi, M., Corpuz, M., Wang, M., Price, M., Andersch, M., Boone, M., Evans, M., Martinez, M., Khona, M., Chrzanowski, M., Lee, M., Dabbah, M., Shoeybi, M., Patwary, M., Mulepati, N., Nabwani, N., Hereth, N., Assaf, N., Habibi, N., Zmora, N., Haber, N., Sessions, N., Bhatia, N., Jukar, N., Pope, N., Ludwig, N., Tajbakhsh, N., Ailon, N., Juluru, N., Sharma, N., Hrinchuk, O., Kuchaiev, O., Delalleau, O., Olabiyi, O., Argov, O. U., Puny, O., Tropp, O., Xie, O., Chadha, P., Shamis, P., Gibbons, P., Molchanov, P., Morkisz, P., Dykas, P., Jin, P., Xu, P., Januszewski, P., Thombre, P. P., Varshney, P., Gundecha, P., Tredak, P., Miao, Q., Wan, Q., Mahabadi, R. K., Garg, R., El-Yaniv, R., Zilberstein, R., Shafipour, R., Harang, R., Izzo, R., Shahbazyan, R., Garg, R., Borkar, R., Gala, R., Islam, R., Hesse, R., Waleffe, R., Watve, R., Koren, R., Zhang, R., Hewett, R., Hewett, R. J., Prenger, R., Timbrook, R., Mahdavi, S., Modi, S., Kriman, S., Lim, S., Kariyappa, S., Satheesh, S., Kaji, S., Pasumarthi, S., Muralidharan, S., Narentharen, S., Narenthiran, S., Bak, S., Kashirsky, S., Poulos, S., Mor, S., Ramasamy, S., Acharya, S., Ghosh, S., Sreenivas, S. T., Thomas, S., Fan, S., Gopal, S., Prabhumoye, S., Pachori, S., Toshniwal, S., Ding, S., Singh, S., Sun, S., Ithape, S., Majumdar, S., Singhal, S., Sergienko, S., Alborghetti, S., Ge, S., Devare, S. D., Barua, S. K., Panguluri, S., Gupta, S., Priyadarshi, S., Akter, S. N., Bui, T., Ene, T.-D., Kong, T., Do, T., Blankevoort, T., Moon, T., Balough, T., Asida, T., Natan, T. B., Ronen, T., Konuk, T., Vashishth, T., Karpas, U., De, U., Noorozi, V., Noroozi, V., Srinivasan, V., Elango, V., Cui, V., Korthikanti, V., Rao, V., Kurin, V., Lavrukhin, V., Anisimov, V., Jiang, W., Ahmad, W. U., Du, W., Ping, W., Zhou, W., Jennings, W., Zhang, W., Prazuch, W., Ren, X., Karnati, Y., Choi, Y., Meyer, Y., Wu, Y.-F., Zhang, Y., Qin, Y., Lin, Y., Geifman, Y., Fu, Y., Subara, Y., Suhara, Y., Gao, Y., Moshe, Z., Dong, Z., Zhu, Z., Liu, Z., Chen, Z., and Yan, Z. Nvidia nemotron 3: Efficient and open intelligence, 2025. URL https://arxiv.org/abs/2512.20856.

Ovadia, Y., Fertig, E., Ren, J., Nado, Z., Sculley, D., Nowozin, S., Dillon, J. V., Lakshminarayanan, B., and Snoek, J. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift, 2019.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Platt, J. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in Large Margin Classifiers*, 10(3), 1999.

Si, C., Gan, Z., Yang, Z., Wang, S., Wang, J., Boyd-Graber, J., and Wang, L. Prompting gpt-3 to be reliable, 2023.

Tan, H., Yang, W., Zhang, J., Chen, X., Tang, R., Su, D., Wang, J., Wang, Y., Sun, F., and Cheng, X. Basecal: Unsupervised confidence calibration via base model signals, 2026. URL https://arxiv.org/abs/2601.03042.

Tian, K., Mitchell, E., Zhou, A., Sharma, A., Rafailov, R., Yao, H., Finn, C., and Manning, C. D. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback, 2023.

Ulmer, D., Gubri, M., Lee, H., Yun, S., and Oh, S. J. Calibrating large language models using their generations only, 2024. URL https://arxiv.org/abs/2403.05973.

Wang, D.-B., Feng, L., and Zhang, M.-L. Rethinking calibration of deep neural networks: Do not be afraid of overconfidence. In Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 11809–11820. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/61f3a6dbc9120ea78ef75544826c814e-Paper.pdf.

Wang, X., Wei, J., Schuurmans, D., Le, Q., Chi, E., Narang, S., Chowdhery, A., and Zhou, D. Self-consistency improves chain of thought reasoning in language models, 2023. URL https://arxiv.org/abs/2203.11171.

Wang, X., Tang, Z., Guo, J., Meng, T., Wang, C., Wang, T., and Jia, W. Empowering edge intelligence: A comprehensive survey on on-device ai models. *ACM Computing Surveys*, 57(9):1–39, April 2025a. ISSN 1557-7341. doi: 10.1145/3724420. URL http://dx.doi.org/10.1145/3724420.

Wang, Y., Zhang, P., Tang, J., Wei, H., Yang, B., Wang, R., Sun, C., Sun, F., Zhang, J., Wu, J., Cang, Q., Zhang, Y., Huang, F., Lin, J., Huang, F., and Zhou, J. Polymath: Evaluating mathematical reasoning in multilingual contexts. *arXiv preprint arXiv:2504.18428*, 2025b. URL https://arxiv.org/abs/2504.18428.

Welbl, J., Liu, N. F., and Gardner, M. Crowdsourcing multiple choice science questions, 2017. URL https://arxiv.org/abs/1707.06209.

Xiao, Y., Liang, P. P., Bhatt, U., Neiswanger, W., Salakhutdinov, R., and Morency, L.-P. Uncertainty quantification with pre-trained language models: A large-scale empirical analysis, 2022.

Xiong, M., Hu, Z., Lu, X., Li, Y., Fu, J., He, J., and Hooi, B. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms, 2024.

Yadkori, Y. A., Kuzborskij, I., Stutz, D., György, A., Fisch, A., Doucet, A., Beloshapka, I., Weng, W.-H., Yang, Y.-Y., Szepesvári, C., Cemgil, A. T., and Tomasev, N. Mitigating llm hallucinations via conformal abstention, 2024.

Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., Zheng, C., Liu, D., Zhou, F., Huang, F., Hu, F., Ge, H., Wei, H., Lin, H., Tang, J., Yang, J., Tu, J., Zhang, J., Yang, J., Yang, J., Zhou, J., Zhou, J., Lin, J., Dang, K., Bao, K., Yang, K., Yu, L., Deng, L., Li, M., Xue, M., Li, M., Zhang, P., Wang, P., Zhu, Q., Men, R., Gao, R., Liu, S., Luo, S., Li, T., Tang, T., Yin, W., Ren, X., Wang, X., Zhang, X., Ren, X., Fan, Y., Su, Y., Zhang, Y., Zhang, Y., Wan, Y., Liu, Y., Wang, Z., Cui, Z., Zhang, Z., Zhou, Z., and Qiu, Z. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Yang, X., Han, J., Bommasani, R., Luo, J., Qu, W., Zhou, W., Bibi, A., Wang, X., Yoon, J., Stengel-Eskin, E., Tong, S., Shen, L., Rafailov, R., Li, R., Wang, Z., Zhou, Y., Cui, C., Wang, Y., Zheng, W., Zhou, H., Gu, J., Chen, Z., Xia, P., Lee, T., Zollo, T., Sehwag, V., Leng, J., Chen, J., Wen, Y., Zhang, H., Deng, Z., Zhang, L., Izmailov, P., Koh, P. W., Tsvetkov, Y., Wilson, A., Zhang, J., Zou, J., Xie, C., Wang, H., Torr, P., McAuley, J., Alvarez-Melis, D., Tramèr, F., Xu, K., Jana, S., Callison-Burch, C., Vidal, R., Kokkinos, F., Bansal, M., Chen, B., and Yao, H. Reliable and responsible foundation models: A comprehensive survey, 2026. URL https://arxiv.org/abs/2602.08145.

Yoon, D., Kim, S., Yang, S., Kim, S., Kim, S., Kim, Y., Choi, E., Kim, Y., and Seo, M. Reasoning models better express their confidence, 2025. URL https://arxiv.org/abs/2505.14489.

Zadrozny, B. and Elkan, C. Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers. *ICML*, 1, 05 2001.

Zadrozny, B. and Elkan, C. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '02, pp. 694–699, New York, NY, USA, 2002. Association for Computing Machinery. ISBN 158113567X. doi: 10.1145/775047.775151. URL https://doi.org/10.1145/775047.775151.

Zhou, K., Jurafsky, D., and Hashimoto, T. Navigating the grey area: How expressions of uncertainty and overconfidence affect language models, 2023.

Zollo, T. P., Deng, Z., Snell, J. C., Pitassi, T., and Zemel, R. Improving predictor reliability with selective recalibration, 2024. URL https://arxiv.org/abs/2410.05407.

A. Additional Experiment Details

Here we report additional experiment details. All models and datasets are available for public download from huggingface.com. We use the vllm (Kwon et al., 2023) library to produce all language model outputs and embeddings. For fair comparison, all models and datasets use the exact same prompt, sampling parameters, and other setup. We sample train data with temperature of 0.7, TopP=0.95, and TopK=20, and test data with temperature of 0.6, TopP=0.95, and TopK=20. Our prompt is:

```
Please reason step by step, and put your final answer within \boxed{}.
Question: {question}
```

For each dataset, we sample 1000 examples. For PolyMath, we sample 125 examples from each of 8 languages (EN, ES, FR, PT, IT, RU, AR, ID); we use the "low" split in order to facilitate easy evaluation via matching.

A.1. Our Algorithm

See Code Box 1 for sample code for training our unsupervised confidence calibrator and applying it to new test inputs.

A.2. Evaluation

For all tasks, we evaluate exact match correctness (robust to simple casing and leading and trailing whitespace). Both TriviaQA and WebQ offer a list of potential matches where appropriate (e.g., names with and without middle initial). ECE1 and ECE2 are calculated using the software package released with (Kumar et al., 2020) (https://github.com/p-lambda/verified_calibration). MCE, Brier score, and AUROC are calculated using the sklearn python package (Pedregosa et al., 2011).

A.3. Baselines

Here we provide more details for baseline implementations.

A.3.1. Token Probabilities

For the token probabilities baselines, we extract top token probabilities using vllm's built-in API. We length-normalize the probabilities by computing their geometric mean, as is typical in language modeling uncertainty research (Malinin & Gales, 2020).

For the Token Probs. baseline we compute this for the entire response produced by the LLM. For the Answer Probs. baseline, we do so for the tokens inside \boxed{}.

A.3.2. Verbalized Confidence

Empirical research has repeatedly shown that when LLMs are asked to verbalize their confidence in a response, the resulting estimates are often poorly calibrated, and good calibration is often incidental (Xiong et al., 2024; Zhou et al., 2023). For example, in recent work, Damani et al. (2025) found that LLMs trained with reinforcement learning perform poorly at verbalizing confidence, usually worse than just using the token probabilities (see Table 1 in the article).

However, in an effort to present the strongest baselines possible, we search for our prompt among 5 prompts written by GPT-5.2 and 5 written by Gemini-3 (both via their respective web interface). We rank all prompts based on our evaluation measures, as well as the rate at which the model fails to produce a well-formatted confidence estimate using the prompt (Imputed, where lower implies better). When the model fails to produce a well-formatted confidence estimate, we impute the estimate as the average estimate across the test set where estimates are provided (also a strong implementation choice for the baseline, as the test set will generally not all be available at once).

We run our optimization across 4 models (Qwen3-1.7B/8B, Nemotron-Cascade-8B-Thinking, deepseek-ai/DeepSeek-R1-Distill-Llama-8B) and 2 datasets (GSM8K, TriviaQA). Results for the 10 prompts are shown in Table 3, where GPT-5.2 produces prompts 0-4, and Gemini-3 produces prompts 5-9. The best-performing prompt, which we use in our experiments, is shown in Prompt Box 2. All 10 prompts can be found in our code.

Code Box 1 Code for training and applying our confidence calibrator model. It requires a train set of input embeddings and self-consistency targets, and produces confidence estimates for new test inputs.

```
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import Ridge
from sklearn.isotonic import IsotonicRegression
from sklearn.model_selection import train_test_split

# 1. Split calibration data into 2 sets for
# training ridge and isotonic regressors
X_a, X_b, y_a, y_b = train_test_split(
    X_train,
    y_train,
    test_size=split_frac,
    random_state=random_state,
)

# 2. Train ridge regression
ridge = Pipeline([
    ("scaler", StandardScaler()),
    ("reg", Ridge(alpha=1.0)),
])
ridge.fit(X_a, y_a)

# 3. Produce inputs to isotonic regression
pred_b = ridge.predict(X_b)

# 4. Train isotonic regression
iso = IsotonicRegression(
    y_min=0.0,
    y_max=1.0,
    out_of_bounds="clip",
)
iso.fit(pred_b, y_b)

# 5. Estimate confidence for test inputs
conf_test = iso.predict(ridge.predict(X_test))
```

Prompt	ECE1	ECE2	MCE	BS	Imputed	AUROC	Rank
6	0.201	0.224	0.389	0.211	0.008	0.649	1
8	0.195	0.221	0.363	0.214	0.009	0.628	2
9	0.204	0.227	0.337	0.216	0.007	0.638	3
4	0.204	0.228	0.344	0.216	0.012	0.629	4.5
5	0.202	0.230	0.355	0.216	0.014	0.642	4.5
0	0.203	0.228	0.340	0.220	0.008	0.624	6
3	0.202	0.226	0.369	0.218	0.016	0.630	7
1	0.205	0.229	0.373	0.215	0.015	0.636	8
2	0.209	0.227	0.346	0.218	0.019	0.621	9
7	0.227	0.249	0.363	0.233	0.013	0.608	10

Table 3. Results for optimization of verbalized confidence baseline.

Prompt Box 2 Verbalized confidence prompt used in our experiments.

```
You are a fact-checking and calibration model.
Task: determine the likelihood that the ANSWER accurately solves the given QUESTION.
Output format (STRICT):
1) Use at most FOUR concise sentences to explain your reasoning.
2) On a NEW LINE, output ONLY the probability using the format \boxed{p} (p is a decimal
    in [0,1]).
   - Format examples: \boxed{0.18}, \boxed{0.42}, \boxed{0.66}, \boxed{0.99}
   - Do not reuse these example scores; instead, be as accurate as possible for this
    specific case.
3) Do NOT write any text after the \boxed{...} line.
```

A.3.3. Supervised Platt Scaling

We perform Platt Scaling using the software package released with (Kumar et al., 2020).

B. Base Models for Weak Supervision

Recent work on unsupervised calibration for LLMs relies on the confidence estimates from a paired pre-trained or base model for calibration (Luo et al., 2025; Tan et al., 2026). This approach is motivated by the common empirical finding that models trained with reinforcement learning display worse calibration than the base models from which they are derived (Kadavath et al., 2022; Leng et al., 2024). However, these methods are not suitable for our setting. (Luo et al., 2025) is focused on multiple choice tasks, and does not study the problem for open-ended generation.

More importantly, these methods both rely on using the base model to assign confidence estimates to the output of a post-trained model. This design may fail conceptually with reasoning models, though: long chain-of-thought outputs and greatly enhanced capabilities can create a substantial distribution shift for the paired base model, meaning their token probabilities may no longer be reflective of task correctness. To verify this hypothesis, we run a set of experiments with all 5 Qwen3 models (and their paired base model versions) where we follow Luo et al. (2025); Tan et al. (2026) in extracting the base model confidence for the output produced by the post-trained model. In order to present the strongest case possible, we explore the method both with (**Base Response Probs.**) and without (**Answer Response Probs.**) reasoning chains included.

Results by model and dataset are presented in Tables 4 and 5. We compare the results to **self-consistency** scores with 100 samples. In our setting, base-model-derived confidence scores fail trivially, performing far worse than self-consistency across all Qwen3 models and datasets. Given the conceptual gap and empirical failure, and our goal to present clear and concise results, we do not include these methods in our main experiments.

Model	Method	ECE2	BS	AUROC
Qwen3-0.6B	Base Answer Probs.	0.410	0.392	0.471
	Base Response Probs.	0.221	0.241	0.390
	Self-consistency	0.103	0.158	0.779
Qwen3-1.7B	Base Answer Probs.	0.568	0.543	0.458
	Base Response Probs.	0.273	0.256	0.344
	Self-consistency	0.112	0.153	0.780
Qwen3-4B-Thinking-2507	Base Answer Probs.	0.635	0.595	0.496
	Base Response Probs.	0.325	0.271	0.410
	Self-consistency	0.095	0.138	0.777
Qwen3-8B	Base Answer Probs.	0.664	0.626	0.482
	Base Response Probs.	0.280	0.244	0.366
	Self-consistency	0.104	0.140	0.777
Qwen3-14B	Base Answer Probs.	0.638	0.574	0.507
	Base Response Probs.	0.291	0.250	0.434
	Self-consistency	0.103	0.134	0.788

Table 4. Comparing potential sources of weak labels for unsupervised calibration; results are by model, averaged across 5 datasets.

Dataset	Method	ECE2	BS	AUROC
GSM8K	Base Answer Probs.	0.807	0.732	0.490
	Base Response Probs.	0.401	0.235	0.385
	Self-consistency	0.057	0.065	0.798
PolyMath	Base Answer Probs.	0.675	0.581	0.544
	Base Response Probs.	0.362	0.237	0.353
	Self-consistency	0.098	0.101	0.876
SciQ	Base Answer Probs.	0.608	0.595	0.397
	Base Response Probs.	0.245	0.287	0.418
	Self-consistency	0.111	0.200	0.725
Trivia QA	Base Answer Probs.	0.479	0.469	0.504
	Base Response Probs.	0.245	0.263	0.357
	Self-consistency	0.076	0.151	0.841
WebQ	Base Answer Probs.	0.346	0.352	0.480
	Base Response Probs.	0.136	0.240	0.429
	Self-consistency	0.168	0.232	0.664

Table 5. Comparing potential sources of weak labels for unsupervised calibration; results are by dataset, averaged across 5 Qwen3 models.

Model	GSM8K	PolyMath	SciQ	Trivia QA	WebQ
Qwen3-0.6B	0.809	0.642	0.409	0.217	0.211
Qwen3-1.7B	0.908	0.839	0.594	0.382	0.353
Qwen3-4B-Thinking-2507	0.949	0.921	0.662	0.543	0.420
Qwen3-8B	0.949	0.922	0.666	0.642	0.444
Qwen3-14B	0.947	0.919	0.692	0.700	0.464
DeepSeek-R1-Distill-Llama-8B	0.872	0.464	-	-	-
DeepSeek-R1-Distill-Qwen-7B	0.915	0.606	-	-	-
OpenReasoning-Nemotron-7B	0.895	0.820	-	-	-
Nemotron-Cascade-8B-Thinking	0.930	0.890	0.668	0.528	0.410

Table 6. Accuracy for all model/dataset combinations. Each row is a model, and our 5 datasets are on the columns.

C. Additional Results

Here we include additional results for all experiments.

C.1. Main Experiment

Accuracy for all model/dataset combinations is reported in Table 6. For more granular results, averages for all datasets and models are presented in Tables 7, 8, and 9. Table 7 features our method, unsupervised and supervised baselines, true test-time self-consistency, as well as ablations for the input embeddings to our model (external model and question text only).

Beyond quantitative results, we include a selection of reliability diagrams produced by our method and baselines. Figures feature results from Qwen3-1.7B (8), Qwen3-14B (9), and OpenReasoning-Nemotron-7B (10).

Dataset	Method	ECE1	ECE2	MCE	BS	AUROC
GSM8K	Token Probs.	0.075	0.083	0.135	0.086	0.655
	Ans. Probs.	0.089	0.107	0.221	0.090	0.615
	Verbal Conf.	0.096	0.130	0.310	0.106	0.638
	Ours	0.045	0.058	0.133	0.078	0.694
	Test-Time SC	0.044	0.057	0.139	0.065	0.798
	Supervised	0.038	0.047	0.101	0.079	0.642
	Ques. only	0.045	0.051	0.101	0.083	0.576
	Black Box	0.045	0.057	0.130	0.077	0.689
PolyMath	Token Probs.	0.130	0.149	0.251	0.169	0.661
	Ans. Probs.	0.205	0.256	0.538	0.204	0.735
	Verbal Conf.	0.175	0.200	0.350	0.194	0.631
	Ours	0.071	0.093	0.209	0.126	0.767
	Test-Time SC	0.067	0.098	0.235	0.101	0.876
	Supervised	0.054	0.068	0.142	0.141	0.665
	Ques. only	0.069	0.082	0.171	0.135	0.697
	Black Box	0.072	0.098	0.226	0.123	0.781
Trivia QA	Token Probs.	0.289	0.322	0.500	0.304	0.701
	Ans. Probs.	0.464	0.477	0.617	0.466	0.556
	Verbal Conf.	0.381	0.407	0.565	0.391	0.670
	Ours	0.080	0.095	0.211	0.184	0.760
	Test-Time SC	0.063	0.076	0.148	0.151	0.841
	Supervised	0.057	0.070	0.142	0.197	0.701
	Ques. only	0.069	0.082	0.192	0.200	0.707
	Black Box	0.078	0.093	0.206	0.187	0.757
SciQ	Token Probs.	0.186	0.214	0.371	0.256	0.656
	Ans. Probs.	0.317	0.332	0.482	0.338	0.567
	Verbal Conf.	0.333	0.345	0.437	0.350	0.616
	Ours	0.057	0.071	0.194	0.209	0.672
	Test-Time SC	0.094	0.111	0.202	0.200	0.725
	Supervised	0.060	0.073	0.145	0.213	0.656
	Ques. only	0.053	0.067	0.182	0.222	0.606
	Black Box	0.057	0.071	0.185	0.209	0.676
WebQ	Token Probs.	0.399	0.409	0.537	0.386	0.604
	Ans. Probs.	0.548	0.557	0.707	0.537	0.574
	Verbal Conf.	0.483	0.510	0.644	0.488	0.621
	Ours	0.106	0.126	0.273	0.230	0.624
	Test-Time SC	0.138	0.168	0.326	0.232	0.664
	Supervised	0.063	0.077	0.151	0.223	0.600
	Ques. only	0.099	0.119	0.261	0.236	0.588
	Black Box	0.105	0.127	0.272	0.233	0.612

Table 7. Averages for all datasets.

Model	Method	ECE1	ECE2	MCE
Qwen3-0.6B	Token Probs.	0.256	0.269	0.374
	Ans. Probs.	0.493	0.519	0.731
	Verbal Conf.	0.476	0.501	0.628
	Ours	0.081	0.096	0.226
Qwen3-1.7B	Token Probs.	0.247	0.269	0.395
	Ans. Probs.	0.362	0.374	0.510
	Verbal Conf.	0.306	0.332	0.470
	Ours	0.081	0.101	0.228
Qwen3-4B-Thinking-2507	Token Probs.	0.176	0.195	0.302
	Ans. Probs.	0.235	0.263	0.445
	Verbal Conf.	0.251	0.268	0.416
	Ours	0.063	0.078	0.180
Qwen3-8B	Token Probs.	0.165	0.184	0.285
	Ans. Probs.	0.250	0.263	0.389
	Verbal Conf.	0.204	0.221	0.344
	Ours	0.071	0.088	0.199
Qwen3-14B	Token Probs.	0.145	0.167	0.302
	Ans. Probs.	0.227	0.246	0.382
	Verbal Conf.	0.183	0.196	0.279
	Ours	0.065	0.081	0.189
OpenReasoning-Nemotron-7B	Token Probs.	0.056	0.066	0.121
	Ans. Probs.	0.125	0.142	0.236
	Verbal Conf.	0.226	0.343	0.788
	Ours	0.064	0.091	0.215
Nemotron-Cascade-8B-Thinking	Token Probs.	0.222	0.241	0.389
	Ans. Probs.	0.292	0.301	0.420
	Verbal Conf.	0.252	0.260	0.331
	Ours	0.064	0.080	0.185
DeepSeek-R1-Distill-Llama-8B	Token Probs.	0.263	0.279	0.393
	Ans. Probs.	0.313	0.353	0.715
	Verbal Conf.	0.231	0.276	0.536
	Ours	0.062	0.074	0.171
DeepSeek-R1-Distill-Qwen-7B	Token Probs.	0.199	0.221	0.326
	Ans. Probs.	0.226	0.320	0.691
	Verbal Conf.	0.177	0.210	0.415
	Ours	0.061	0.078	0.173

Table 8. Averages for all models, ECE and MCE.

Model	Method	BS	AUROC
Qwen3-0.6B	Token Probs.	0.266	0.640
	Ans. Probs.	0.481	0.621
	Verbal Conf.	0.482	0.592
	Ours	0.187	0.654
Qwen3-1.7B	Token Probs.	0.265	0.692
	Ans. Probs.	0.362	0.584
	Verbal Conf.	0.317	0.637
	Ours	0.173	0.699
Qwen3-4B-Thinking-2507	Token Probs.	0.203	0.661
	Ans. Probs.	0.253	0.596
	Verbal Conf.	0.259	0.621
	Ours	0.151	0.705
Qwen3-8B	Token Probs.	0.197	0.699
	Ans. Probs.	0.253	0.618
	Verbal Conf.	0.221	0.684
	Ours	0.148	0.715
Qwen3-14B	Token Probs.	0.182	0.708
	Ans. Probs.	0.232	0.641
	Verbal Conf.	0.202	0.671
	Ours	0.147	0.698
OpenReasoning-Nemotron-7B	Token Probs.	0.120	0.606
	Ans. Probs.	0.139	0.446
	Verbal Conf.	0.235	0.571
	Ours	0.109	0.788
Nemotron-Cascade-8B-Thinking	Token Probs.	0.246	0.657
	Ans. Probs.	0.294	0.612
	Verbal Conf.	0.262	0.621
	Ours	0.155	0.719
DeepSeek-R1-Distill-Llama-8B	Token Probs.	0.263	0.525
	Ans. Probs.	0.301	0.718
	Verbal Conf.	0.254	0.604
	Ours	0.152	0.723
DeepSeek-R1-Distill-Qwen-7B	Token Probs.	0.205	0.534
	Ans. Probs.	0.217	0.825
	Verbal Conf.	0.189	0.694
	Ours	0.127	0.751

Table 9. Averages for all models, Brier score and AUROC.

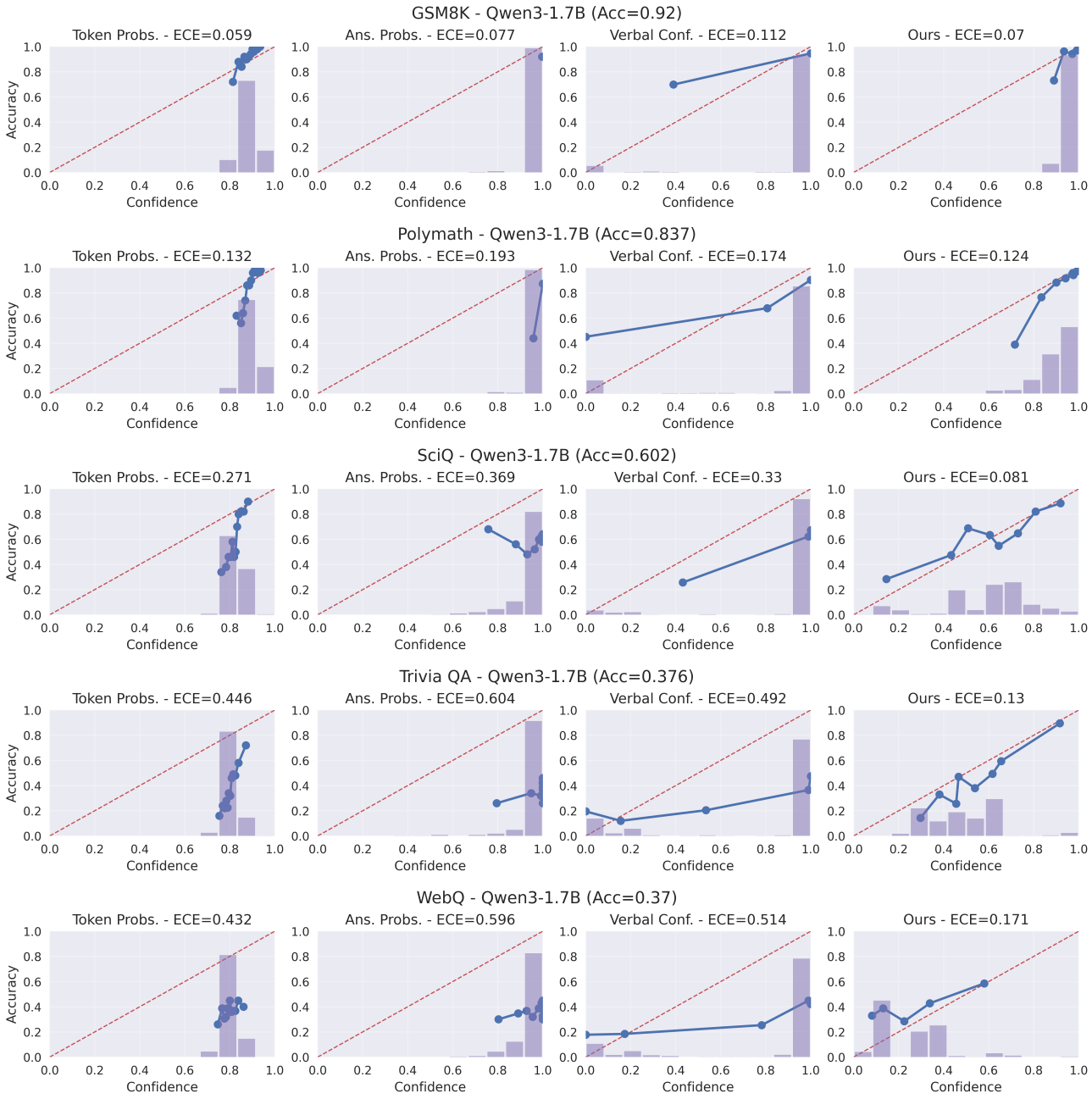

Figure 8. Reliability diagrams for Qwen3-1.7B.

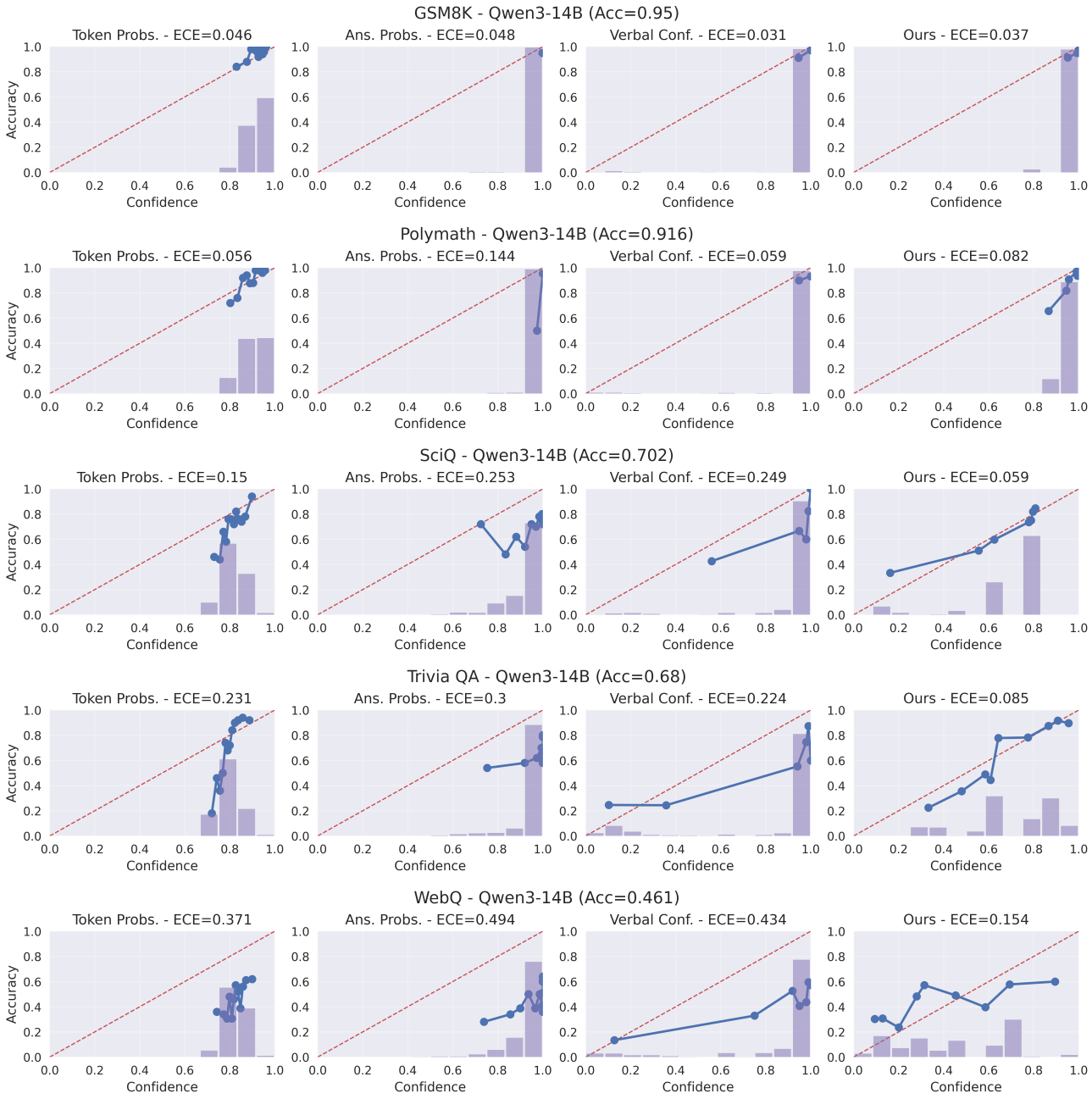

Figure 9. Reliability diagrams for Qwen3-14B.

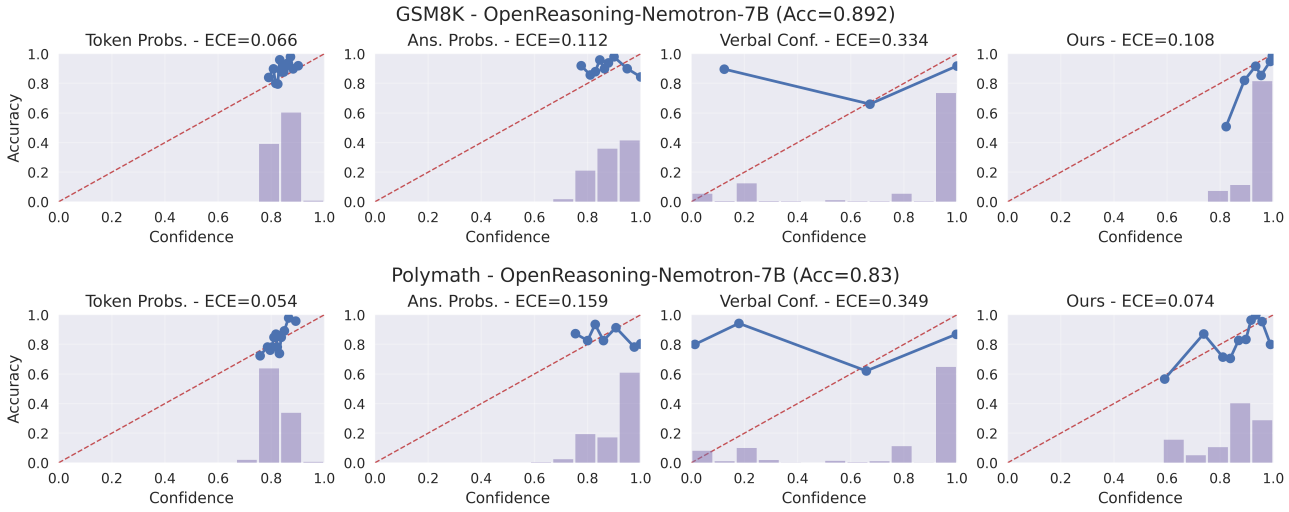

Figure 10. Reliability diagrams for OpenReasoning-Nemotron-7B.

k	ECE2	BS	AUROC
5	0.0975	0.1625	0.6816
10	0.0887	0.1580	0.6950
20	0.0875	0.1563	0.7021
50	0.0869	0.1552	0.7068
100	0.0864	0.1550	0.7080
Best Unsup.	0.2155	0.2214	0.6560

Table 10. Ablation on the number of samples (k) produced for computing calibration targets. For each level of k, we run 100 random trials in the same fashion as our main experiment. We include results for the best unsupervised baseline (Token Probs.) for easy comparison.

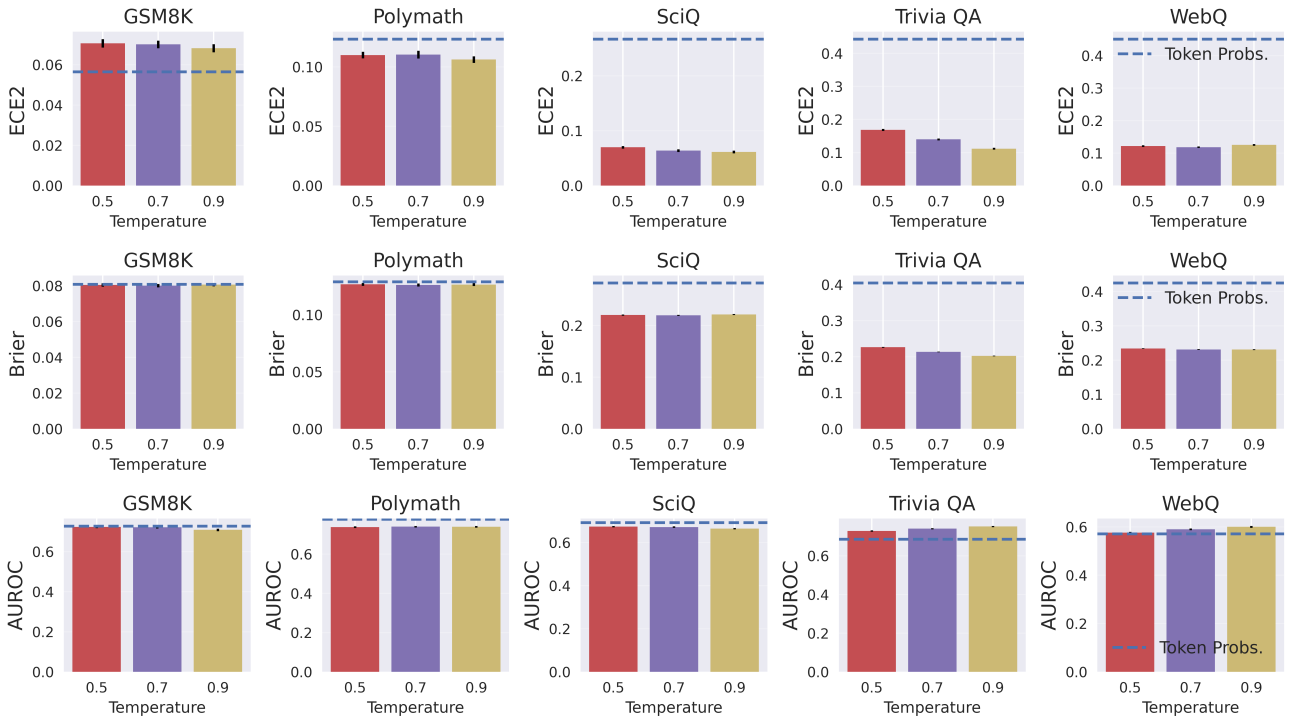

Figure 11. Temperature ablations for Qwen3-1.7B.

C.1.1. Additional Ablations

In Table 10, we ablate the number of samples used to produce the proxy targets. For each of $k \in \{5, 10, 20, 50\}$, we sample k train examples over 20 random trials, and compare to the results produced using 100 samples. We run the study over all models and datasets, making the results comparable to those in Figure 2. We find that our method easily beats the best unsupervised baseline (Token Probs.) with as few as 5 samples, but continues to improve up to our largest setting of $k = 100$.

In Figures 11 and 12, we ablate the temperature used in sampling train data (with $k = 20$ samples) for Qwen3-1.7B and -4B and all 5 datasets. We compare sampling with temperatures 0.5 and 0.9 to our choice of 0.7. We find our choice to be best, although our method easily outperforms the Token Probs. baseline with any of these settings.

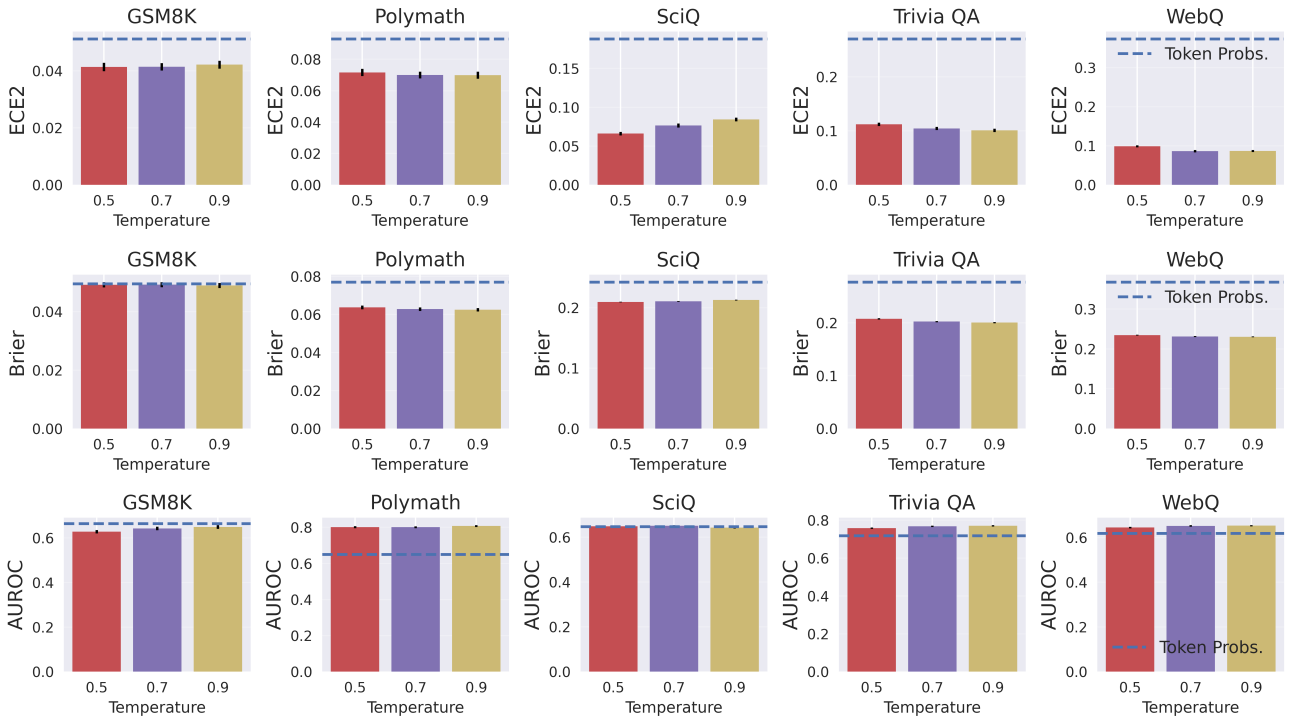

Figure 12. Temperature ablations for Qwen3-4B-Thinking-2507.

C.2. Distribution Shifts

We explore 4 language splits for the PolyMath OOD experiments, outlined below. For each split, we use each language group as both train and test set, for a total of 8 experiments. See Table 11 for results by split.

- **Latin-script-oriented split:** partitions the languages into a Western European, Latin-script-heavy group (English, Spanish, French, Portuguese) and a more typologically and script-diverse group (Italian, Russian, Arabic, Indonesian).
- **High-resource split:** partitions the languages into a higher-resource group (English, Spanish, French, Russian) and a more mixed-resource group (Portuguese, Italian, Arabic, Indonesian).
- **Script-based split:** partitions the languages into a script-diverse group including non-Latin scripts (English, Spanish, Russian, Arabic) and an all-Latin-script group (French, Portuguese, Italian, Indonesian).
- **Romance split:** partitions the languages into a Romance-language group (Spanish, French, Portuguese, Italian) and a complementary non-Romance group (English, Russian, Arabic, Indonesian).

We also present full results for calibration under task distribution shift. Table 12 offers results for the QA domain shifts, and Table 13 contains results for the math shifts.

Train Languages	Test Languages	Method	ECE2	BS
EN, ES, FR, PT	IT, RU, AR, ID	Ans. Probs.	0.265	0.211
		Token Probs.	0.161	0.177
		Verbal Conf.	0.196	0.194
		Ours	0.102	0.136
IT, RU, AR, ID	EN, ES, FR, PT	Ans. Probs.	0.218	0.170
		Token Probs.	0.153	0.154
		Verbal Conf.	0.169	0.169
		Ours	0.074	0.109
EN, ES, FR, RU	PT, IT, AR, ID	Ans. Probs.	0.279	0.216
		Token Probs.	0.168	0.183
		Verbal Conf.	0.196	0.195
		Ours	0.113	0.135
PT, IT, AR, ID	EN, ES, FR, RU	Ans. Probs.	0.195	0.164
		Token Probs.	0.142	0.148
		Verbal Conf.	0.171	0.167
		Ours	0.064	0.110
EN, ES, RU, AR	FR, PT, IT, ID	Ans. Probs.	0.273	0.205
		Token Probs.	0.169	0.184
		Verbal Conf.	0.195	0.192
		Ours	0.109	0.128
FR, PT, IT, ID	EN, ES, RU, AR	Ans. Probs.	0.204	0.174
		Token Probs.	0.127	0.145
		Verbal Conf.	0.169	0.169
		Ours	0.083	0.122
EN, RU, AR, ID	ES, FR, PT, IT	Ans. Probs.	0.262	0.197
		Token Probs.	0.168	0.178
		Verbal Conf.	0.191	0.187
		Ours	0.102	0.123
ES, FR, PT, IT	EN, RU, AR, ID	Ans. Probs.	0.211	0.182
		Token Probs.	0.132	0.152
		Verbal Conf.	0.172	0.175
		Ours	0.091	0.126

Table 11. Results for distribution shift experiments by language split.

Train Domain	Test Domain	Method	ECE2	BS	AUROC
SciQ	Trivia QA	Token Probs.	0.265	0.271	0.725
		Ans. Probs.	0.380	0.375	0.550
		Verbal Conf.	0.310	0.301	0.714
		Ours	0.124	0.240	0.602
SciQ	WebQ	Token Probs.	0.395	0.390	0.623
		Ans. Probs.	0.504	0.490	0.579
		Verbal Conf.	0.456	0.438	0.641
		Ours	0.170	0.266	0.609
Trivia QA	SciQ	Token Probs.	0.188	0.241	0.653
		Ans. Probs.	0.274	0.288	0.586
		Verbal Conf.	0.292	0.297	0.634
		Ours	0.105	0.209	0.675
Trivia QA	WebQ	Token Probs.	0.395	0.390	0.623
		Ans. Probs.	0.504	0.490	0.579
		Verbal Conf.	0.456	0.438	0.641
		Ours	0.141	0.253	0.627
WebQ	SciQ	Token Probs.	0.188	0.241	0.653
		Ans. Probs.	0.274	0.288	0.586
		Verbal Conf.	0.292	0.297	0.634
		Ours	0.154	0.221	0.689
WebQ	Trivia QA	Token Probs.	0.265	0.271	0.725
		Ans. Probs.	0.380	0.375	0.550
		Verbal Conf.	0.310	0.301	0.714
		Ours	0.107	0.202	0.750

Table 12. Distribution shift results for QA domains.

Train Domain	Test Domain	Method	ECE2	BS	AUROC
GSM8K	PolyMath	Token Probs.	0.144	0.165	0.640
		Ans. Probs.	0.251	0.190	0.744
		Verbal Conf.	0.181	0.181	0.622
		Ours	0.138	0.152	0.677
PolyMath	GSM8K	Token Probs.	0.082	0.078	0.640
		Ans. Probs.	0.091	0.076	0.603
		Verbal Conf.	0.120	0.095	0.631
		Ours	0.053	0.069	0.694

Table 13. Distribution shift results for math domains.

Prompt Box 3 System prompt for GPT-4o-mini when using the answer and confidence score from the assistant model.

```
You are acting as a downstream decision maker being advised by an LLM that outputs an
    answer and its confidence score.

You will be given:
(1) a question,
(2) a candidate answer produced by an upstream system,
(3) a numeric confidence score in [0,1] from that upstream system.

CRITICAL INSTRUCTIONS:
- Treat the provided confidence score as probabilistically informative: interpret it as
    the probability that the candidate answer is correct.
- Do not ignore the candidate answer + the numeric score. Do NOT solve the problem from
    scratch without considering them.
- Still make your own assessment, and output a new answer and confidence score based on
    your knowledge.
- The final_answer string should be a short answer (e.g., the name of a single entity)

Output exactly:
- final_answer (string)
- final_confidence (number in [0,1])
```

Prompt Box 4 Per-example prompt for GPT-4o-mini when using the answer and confidence score from the assistant model.

```
Question:
{question}

Candidate answer:
{candidate_answer}

Upstream confidence score '{method_name}' in [0,1]: {score}
```

Prompt Box 5 Baseline prompt for GPT-4o-mini without any input from an assistant.

```
You are answering questions.

Return:
- answer: a short final answer (no explanation; just the answer).
- confidence: a number in [0,1] representing the probability that your answer is correct.

Be honest and well-calibrated. If you are unsure, lower the confidence.
```

C.3. Linguistic Calibration

This section contains additional details and results for our linguistic calibration experiments. The system and per-example prompts for GPT-4o-mini are given in Prompt Boxes 3 and 4. The baseline prompt without proposed answer and confidence is in Prompt Box 5.

Table 14 contains full results for the experiments.

Model	Method	Acc.	ECE2	MCE	BS
Qwen3-0.6B	Token Probs.	0.641	0.164	0.400	0.233
	Verbal Conf.	0.612	0.336	0.350	0.347
	Ours	0.634	0.061	0.100	0.223
Qwen3-1.7B	Token Probs.	0.640	0.255	0.472	0.267
	Verbal Conf.	0.632	0.343	0.630	0.330
	Ours	0.643	0.134	0.203	0.219
Qwen3-4B-Thinking-2507	Token Probs.	0.667	0.222	0.404	0.249
	Verbal Conf.	0.653	0.300	0.476	0.299
	Ours	0.663	0.119	0.400	0.209
Qwen3-8B	Token Probs.	0.667	0.214	0.360	0.248
	Verbal Conf.	0.662	0.315	0.456	0.292
	Ours	0.662	0.160	0.238	0.220
Qwen3-14B	Token Probs.	0.680	0.200	0.350	0.238
	Verbal Conf.	0.670	0.302	0.495	0.281
	Ours	0.673	0.155	0.283	0.219
Average	Token Probs.	0.659	0.211	0.397	0.247
	Verbal Conf.	0.646	0.319	0.482	0.310
	Ours	0.655	**0.126**	**0.245**	**0.218**
Base Model	-	**0.686**	0.275	0.372	0.261

Table 14. Simulated decision making by GPT-4o-mini, full results.