Entropy Sentinel: Continuous LLM Accuracy Monitoring from Decoding Entropy Traces in STEM

Anonymous ACL submission

Abstract

Deploying LLMs raises two coupled challenges: (1) monitoring—estimating where a model underperforms as traffic and domains drift—and (2) improvement—prioritizing data acquisition to close the largest performance gaps. Both require the same prerequisite: a cheap, continuous estimate of per-slice accuracy under domain shift. We test whether inference-time entropy signals can provide this estimate. For each response, we compute an output-entropy profile from final-layer next-token probabilities (from top-k logprobs) and summarize it with different statistics. A lightweight classifier predicts instance correctness, and averaging predicted probabilities yields a domain-level accuracy estimate. We evaluate on ten STEM reasoning benchmarks with exhaustive train/test compositions ($k \in \{1, 2, 3, 4\}$; all $\binom{10}{k}$ combinations), on different classifier models and features across nine LLMs from six families (3B–20B). Estimates often track held-out benchmark accuracy, and several models show near-monotonic ordering of domains, providing evidence that output-entropy profiles are an accessible signal for scalable accuracy monitoring. We found that the dominant design factor is the composition of supervision data: difficulty-spanning training sets consistently outperform homogeneous ones across all models and configurations. [1]

1 Introduction

Deployed LLMs serve heterogeneous traffic that shifts over time. Yet practitioners still lack scalable answers to two tightly coupled questions: *where is the model underperforming on current usage*, and *what data should we acquire to close those gaps?* Both questions share a common bottleneck: reliable, continuous estimation of per-slice accuracy across domains. In practice, this estimation is addressed with manually curated benchmarks and periodic human-labeled evaluations. While effective, this workflow is expensive, slow, and poorly matched to production: coverage across domains and difficulty regimes is incomplete, and it is hard to evaluate continuously at the granularity practitioners care about (e.g., per traffic slice, customer segment, or topic cluster). As a result, teams often discover failures late and collect training data opportunistically rather than targeting the largest performance gaps.

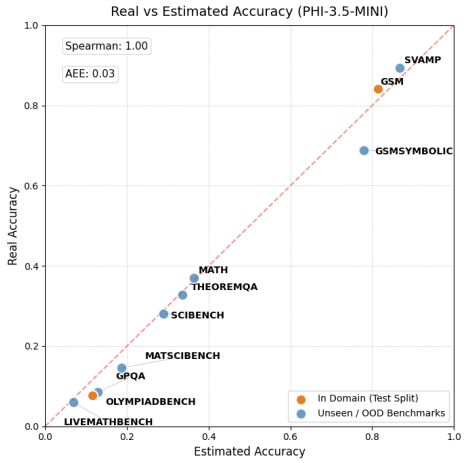

Figure 1: Entropy-based accuracy estimation for PHI-3.5-MINI-3.6B. Trained on two benchmarks (orange), the estimator generalizes to eight unseen STEM benchmarks (blue)

A natural alternative is to rely on signals already produced during inference. If an inexpensive uncertainty trace could *robustly* predict correctness, then we could estimate accuracy for arbitrary slices of production traffic without repeated labeling. Once such estimates are available, they immediately induce a ranking of slices by predicted performance, which could guide downstream decisions such as data-acquisition prioritization. For the underlying signal to be useful in deployment, it must be (i) cheap to extract at scale, (ii) available for both open and closed models (e.g., via top-k

[1] Code for reproducing our experiments is available here.

log-probabilities), and (iii) robust to domain shift, so that supervision from a small set of labeled tasks transfers to unseen ones.

Crucially, monitoring requires a quantity in *accuracy units*—not only a relative uncertainty score. Entropy- and logprob-based traces carry meaningful uncertainty signals (Malinin and Gales, 2021; Kuhn et al., 2023; Bouchard and Chauhan, 2025a; Kadavath et al., 2022), but raw scores are typically not directly interpretable as accuracy and can vary in scale across models and domains. We therefore cast monitoring as predicting correctness probabilities from decoding traces and aggregating them into domain-level accuracy estimates, yielding outputs that are directly actionable for both performance tracking and slice prioritization.

We study whether these noisy but informative signals can be turned into such estimates. Using only top-k decoding log-probabilities, we summarize each generation's entropy trajectory into a compact feature vector and train a lightweight probabilistic correctness predictor. Averaging predicted correctness probabilities over a slice yields an accuracy estimate, enabling a simple deployment primitive: monitor slice accuracy from inference logs and rank slices by estimated performance.

We evaluate in a controlled STEM reasoning setting with ten benchmarks spanning elementary math, advanced math, and science. To stress-test robustness under domain shift, we exhaustively vary supervision: for each $k \in \{1, 2, 3, 4\}$ we train on all $\binom{10}{k}$ benchmark subsets and estimate accuracy on the remaining $10 - k$. We repeat this across nine LLMs from six families (3B–20B) and multiple estimator variants (classifier family, calibration, class balancing and feature set), totaling $> 160{,}000$ configurations. The resulting estimates often track held-out benchmark accuracy closely and achieve high rank agreement, with some LLMs presenting almost perfect calibration on sensible configurations (Figure 1). We nevertheless found that reliability varies across models and should be validated for the target serving model before deployment.

Overall, our results yield two findings: (1) lightweight classifiers trained on entropy profiles provide a practical signal for continuous accuracy monitoring, and (2) the dominant design factor is supervision composition—training sets that span difficulty generalize substantially better than homogeneous ones, a result that holds across all nine LLMs and dominates classifier architecture, feature selection, and calibration choices. We intentionally establish this in a STEM setting with verifiable correctness, enabling a thorough robustness study under a fixed protocol—exhaustive train/test benchmark compositions, multiple estimator and feature variants, and multiple model families. Although our experiments use open-weight models, we restrict ourselves to top-k decoding logprob signals (top-20) exposed by common model-serving APIs, keeping the approach compatible with both open and closed deployments. Extending to less verifiable, open-ended domains, validating transfer to closed commercial models, and closing the loop to downstream data acquisition are natural next steps.

2 From Signatures to Accuracy Estimates

Previous work has shown that token-probability traces carry information about response correctness (Malinin and Gales, 2021; Kuhn et al., 2023; Ali et al., 2025; Bouchard and Chauhan, 2025a). We build on this finding and propose a two-stage methodology for domain-level accuracy monitoring: (1) extract a compact uncertainty feature vector from the decoding trace of each response, and (2) train a lightweight probabilistic classifier to predict instance-level correctness, whose predictions are aggregated into slice-level accuracy estimates.

Entropy from top-k log-probabilities. For an input prompt q, let the model generate an output $\hat{y} = (y_1, \ldots, y_T)$. At decoding step t, let $p^{(t)}(\cdot)$ denote the next-token distribution conditioned on the prompt and previously generated tokens $(q, y_{<t})$. Many APIs expose only top-k next-token probabilities at each step. We therefore approximate entropy by truncating the sum to the top-k tokens: $\tilde{H}^{(t)} = -\sum_{i \in \text{Top-}k} p_i^{(t)} \log p_i^{(t)}$, which differs from the true Shannon entropy because it omits the probability mass outside the Top-k set. We use $\tilde{H}^{(t)}$ as an uncertainty signal over the generated output tokens.

From instance correctness to domain accuracy. Given a response $x = (q, \hat{y})$, we extract a feature vector from its entropy trajectory by summarizing $\{\tilde{H}^{(t)}\}_{t=1}^{T}$ with a small set of statistics (Sec. 3), and train a probabilistic classifier that outputs an estimated probability of correctness $\hat{P}(x) \in [0, 1]$. For a domain (or slice) D represented by a set of instances X_D, we estimate its accuracy by averaging predicted correctness probabilities:

$$\hat{A}(D) = \frac{1}{|X_D|} \sum_{x \in X_D} \hat{P}(x). \qquad (1)$$

If $\hat{P}(x)$ is well-calibrated, then $\hat{A}(D)$ is a consistent estimator of the true domain accuracy, making it suitable for continuous monitoring over production traffic partitioned into slices.

Design requirements. For practical monitoring, the signal and estimator should be: (i) **computationally lightweight** (single-pass, no auxiliary models), (ii) **API-compatible** (requiring only top-k token probabilities, not hidden states or full-vocabulary distributions), and (iii) **robust to domain shift** (transferring from a small labeled source to unseen domains). In Section 3 we identify a compact feature space derived from the entropy trajectory that satisfies (i) and (ii), and empirically validate that it carries instance-level correctness signal. In Section 4 we describe the evaluation protocol designed to stress-test (iii) under different LLMs and configurations, and in Section 5 we present the results.

3 Entropy Profile Signals

To satisfy requirements (i) efficiency and (ii) API compatibility, we restrict ourselves to uncertainty signals available from standard decoding logs and evaluate whether the output-entropy trajectory contains enough signal to discriminate correct from incorrect responses using only compact summaries. We approximate entropy from top-k log-probabilities, setting k=20 to match the maximum exposed by commercial APIs (see Appendix B for an ablation against the full distribution, which shows little difference in predictive power).

Do simple summaries carry correctness signal? For each response, we compute an entropy trajectory over decoding steps, $\{\tilde{H}^{(t)}\}_{t=1}^{T}$. Following standard evaluation practice in uncertainty quantification (Malinin and Gales, 2021; Kuhn et al., 2023; Bouchard and Chauhan, 2025a; Kadavath et al., 2022), we test whether *single-number* summaries of this trajectory already discriminate correct from incorrect outputs. Concretely, for each summary statistic $s(\{\tilde{H}^{(t)}\})$, we treat larger values as a higher score for incorrectness and report its AUROC. We also include traditional white-box uncertainty metrics computed from token log-probabilities (SEA, $\mathrm{NLL}_{\mathrm{avg}}$, $\mathrm{NLL}_{\mathrm{max}}$, $\mathrm{NLL}_{\mathrm{sum}}$, LNTP, MTP, and PPL), defined in Appendix A.

Table 1 reports AUROC for three representative model–benchmark pairs under inference at temperature $T = 0.5$, motivated by recommendations from Kuhn et al. (2023) (additional models and a temperature sensitivity analysis is presented in Appendices C and D respectively). Since each benchmark provides reference answers, we obtain ground-truth correctness labels for every generation and use them to evaluate how well each summary statistic separates correct from incorrect responses. The vast majority of statistics achieve AUROC > 0.5, confirming non-trivial separability: correct responses tend to concentrate in lower-entropy regions, reflecting higher model confidence during decoding, while incorrect responses shift toward higher entropy values, indicating greater token-level uncertainty (Figure 2). This is consistent with previous findings and causes many of these statistics to behave as natural uncertainty scores, where simple thresholding yields better-than-chance discrimination.

Statistic	MATH	GSM8K	OLYMP.
PHI-3.5-MINI 3.6B			
Mean	0.7283	0.7391	0.7269
Std Dev	0.7635	0.7397	0.7456
Max	0.7952	0.7074	0.7778
Q10	**0.8248**	0.7102	0.7589
Q25	0.8084	0.7318	0.7323
Q50	0.7357	0.7412	0.7028
Q75	0.6939	0.7341	0.7027
Q90	0.7130	0.7257	0.7280
Skewness	0.5961	0.6902	0.6603
Kurtosis	0.5726	0.6722	0.6497
SEA	0.8184	**0.7649**	**0.7958**
$\mathrm{NLL}_{\mathrm{avg}}$	0.6983	0.7180	0.7188
$\mathrm{NLL}_{\mathrm{max}}$	0.6937	0.6751	0.6768
$\mathrm{NLL}_{\mathrm{sum}}$	0.8087	0.7497	0.7908
LNTP	0.6983	0.7180	0.7188
MTP	0.6937	0.6751	0.6768
PPL	0.6983	0.7180	0.7188
MINISTRAL-3 8B			
Mean	0.7654	**0.7974**	0.7981
Std Dev	0.7897	0.7970	0.7903
Max	0.7959	0.7681	0.7712
Q10	0.8034	0.7746	0.7943
Q25	0.7933	0.7789	0.8033
Q50	0.7578	0.7968	0.8108
Q75	0.7410	0.7923	0.7988
Q90	0.7598	0.7891	0.7897
Skewness	0.7094	0.7596	0.7906
Kurtosis	0.7053	0.7441	0.7855
SEA	**0.8762**	0.7930	0.8559
$\mathrm{NLL}_{\mathrm{avg}}$	0.6660	0.7086	0.7690
$\mathrm{NLL}_{\mathrm{max}}$	0.5666	0.5233	0.6147
$\mathrm{NLL}_{\mathrm{sum}}$	0.8742	0.7538	**0.8600**
LNTP	0.6660	0.7086	0.7690
MTP	0.5666	0.5233	0.6147
PPL	0.6660	0.7086	0.7690
GPT-OSS 20B			
Mean	0.7736	0.7445	0.8583
Std Dev	0.7771	0.7472	0.8498
Max	0.8137	0.6991	0.7405
Q10	0.7293	0.7243	0.8479
Q25	0.7449	0.7177	0.8437
Q50	0.7454	0.7233	0.8421
Q75	0.7702	0.7476	0.8600
Q90	0.7843	0.7483	0.8643
Skewness	0.7676	0.7448	0.8558
Kurtosis	0.7749	0.7506	0.8607
SEA	**0.9137**	**0.7886**	**0.8953**
$\mathrm{NLL}_{\mathrm{avg}}$	0.4127	0.3333	0.7641
$\mathrm{NLL}_{\mathrm{max}}$	0.5351	0.5206	0.4042
$\mathrm{NLL}_{\mathrm{sum}}$	0.9070	0.7864	0.8869
LNTP	0.4127	0.3333	0.7641
MTP	0.5351	0.5206	0.4042
PPL	0.4127	0.3333	0.7641

Table 1: AUROC of entropy-profile summary stats across 3 benchmarks and 4 models (reporting $1 -$ AUROC for skewness, kurtosis, LNTP, and MTP).

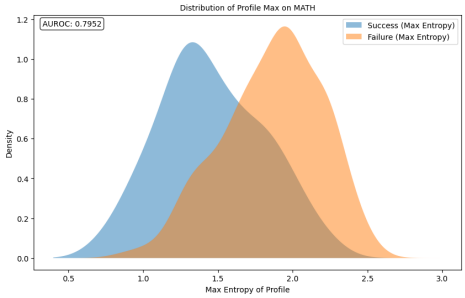

Figure 2: Max-entropy density for PHI-3.5-MINI on MATH (correct vs. incorrect). Incorrect responses shift to higher entropy, indicating greater uncertainty.

However, no single summary is reliably best: the top feature shifts across both models and benchmarks (Table 1), suggesting that different aspects of the entropy profile capture complementary uncertainty cues. For example, lower-tail quantiles (Q10/Q25) are most predictive on MATH, whereas dispersion (Std) is more informative on GSM8K. Accumulation metrics (SEA, NLL_{sum}) consistently rank among the strongest individual signals, while higher-order moments (skewness, kurtosis) help in some settings but collapse toward chance in others. This variability suggests that relying on any single metric risks missing domain- and model-specific failure modes, motivating our use of a compact multi-statistic profile.

A compact discriminative signature. Motivated by these observations, we avoid committing to a single "best" scalar. Instead, we encode each response with a fixed 17D entropy-profile vector capturing central tendency and dispersion (max, mean, std), distributional tails (Q10–Q90), shape (skewness, kurtosis), accumulation (SEA), and the traditional white-box UQ metrics presented in Appendix A, truncating any sums over the logprob vocabulary to the top $k = 20$ tokens where applicable. A lightweight probabilistic classifier can then learn which aspects matter for correctness in a given setting, without brittle manual feature selection.

4 Robustness Evaluation Protocol

We now evaluate whether the methodology supports domain-level accuracy estimation under domain shift: we train instance-level correctness predictors on a small set of labeled benchmarks and estimate accuracy on unseen benchmarks by aggregating predicted correctness probabilities.

Benchmarks. We evaluate on ten STEM reasoning benchmarks spanning math and science: GSM8K (Cobbe et al., 2021), SVAMP (Patel et al., 2021), GSM-Symbolic (Mirzadeh et al., 2025), MATH (Hendrycks et al., 2021), TheoremQA (Chen et al., 2023), SciBench (Wang et al., 2023), MatSciBench (Zhang et al., 2025b), OlympiadBench (He et al., 2024), LiveMathBench (Anonymous, 2025), and GPQA (Rein et al., 2023). All tasks use zero-shot chain-of-thought prompting with free-form final answers. For benchmarks originally multiple-choice (GPQA, SciBench), we remove answer options from the prompt. For OlympiadBench we restrict to text-only math and physics questions; for LiveMathBench we use the `v202505_all_en` subset.[2] When a split exists, we evaluate on the test portion; otherwise we evaluate on the full benchmark.

Instance labeling and prediction target. Each benchmark instance provides a question q, a reference answer $y^\star$, and a model-generated response $\hat{y}$. We extract the model's final answer from $\hat{y}$ using benchmark-specific post-processing (e.g., stripping formatting and selecting the last boxed/numeric expression when applicable). An external *validator* LLM (GROK-4.1-FAST-REASONING (xAI, 2025)) receives $(q, \hat{y}_{\text{final}}, y^\star)$ and outputs a binary label $z \in \{0, 1\}$ indicating whether the final answer matches the reference; a manual audit of 1000 randomly sampled instances yielded 99% agreement with human judgment. We treat z as supervision for our probabilistic correctness predictor.

Models. We evaluate nine LLMs (3B–20B, six families), restricting features to top-20 decoding logprobs to match a common interface constraint in logprob-returning serving stacks. We run the full pipeline separately for Ministral-3 3B (Mistral AI, 2025), Phi-3.5-Mini 3.8B (Microsoft, 2024), Qwen-3 4B (Qwen Team, 2025), Gemma-3 4B (Google DeepMind, 2025), Qwen-3 8B, Ministral-3 8B, Llama-3.1 8B (Meta AI, 2024), Gemma-3 12B, and GPT-OSS 20B (OpenAI, 2025).

Train/test sweep for domain shift. To avoid conclusions tied to a single split, we vary which benchmarks provide supervision. For each $k \in \{1, 2, 3, 4\}$ and each benchmark subset G of size k (in total $\sum_{k=1}^{4} \binom{10}{k} = 385$ groups), we train a correctness predictor on instances from G and evaluate accuracy estimation on the remaining $10 - k$ benchmarks. This yields an OOD setting where all test domains are disjoint from the supervision set.

Baselines The individual UQ metrics from Appendix A return unbounded uncertainty scores that

[2] https://huggingface.co/datasets/opencompass/LiveMathBench

cannot directly serve as accuracy estimates. To create a comparable baseline, we calibrate each metric via Platt scaling (training a logistic regression) over the training set, mapping its scores to correctness probabilities that can be aggregated via Eq. 1. This yields a single-feature, logistic regression variant of our pipeline for each metric.

Estimator configurations. We evaluate three classifier families: logistic regression, random forest, and a multilayer perceptron (MLP). For each classifier, we select hyperparameters via cross-validated grid search on the training group, optimizing ROC-AUC. We also vary two training choices: class balancing (on/off) and isotonic calibration (on/off).

To test whether the full 17D vector is necessary or whether simpler summaries suffice, we train each configuration over four feature subsets of decreasing size: (i) the full 17D profile, (ii) the 10 entropy-distribution statistics only, (iii) max, SE_{sum}, and NLL_{sum}—the three most consistently discriminative signals from the AUROC analysis in Section 3—and (iv) SE_{sum} alone, the single strongest individual metric.

Across 9 models, 385 train/test groups, 3 classifier families, 2×2 training options, 4 feature subsets, and calibrated single-metric baselines, this produces over 160,000 configurations.

Domain-level evaluation metrics. For each held-out benchmark D, we estimate accuracy by averaging per-instance correctness probabilities (Eq. 1). We report: (i) accuracy estimation error **(AEE)**, the mean absolute error between estimated and true benchmark accuracies over held-out domains; and (ii) **Spearman correlation** ρ between estimated and true accuracies, capturing whether the estimator correctly ranks domains for data-acquisition prioritization.

Implementation. All LLMs are served with vLLM (Kwon et al., 2023) (seed 42) with a maximum generation length of 2,048 tokens and temperature 0.5, as recommended for Shannon entropy estimation in Kuhn et al. (2023). Classifiers are implemented in scikit-learn (Pedregosa et al., 2011); features are z-scored using training-group statistics. Further reproducibility details are in Appendix E.

5 Results

Research Questions. We organize results around four questions: **(RQ1)** does the entropy-profile signal support cross-domain accuracy estimation under plausible defaults? **(RQ2)** how does it compare with simple baselines? **(RQ3)** how sensitive is performance to which benchmarks are used for supervision? **(RQ4)** how sensitive is performance to estimator design choices (classifier family, calibration, balancing, feature subset)?

	Extremes		Intermediate	
Model	**AEE**	ρ	**AEE**	ρ
PHI-3.5-MINI (3.6B)	**0.03**	**1.00**	0.06	0.95
MINISTRAL3 (3B)	0.06	0.96	0.11	0.79
MINISTRAL3 (8B)	0.07	0.96	0.12	0.89
QWEN3 (4B)	0.08	0.95	0.11	0.94
QWEN3 (8B)	0.12	0.76	0.17	0.75
GEMMA3 (4B)	0.09	0.94	0.14	0.92
GEMMA3 (12B)	0.08	0.92	0.12	0.85
LLAMA-3.1 (8B)	0.07	0.95	0.11	0.92
GPT-OSS (20B)	0.15	0.90	0.16	0.89

Table 2: Cross-domain accuracy estimation with two *a priori* training sets—*Extremes* (GSM8K+OlympiadBench) and *Intermediate* (MATH+SciBench)—using a calibrated, class-balanced random forest. ρ: Spearman correlation.

Method	**Spearman** $\rho \uparrow$	**AEE** $\downarrow$
Default (RQ1 Config)	$\underline{.95}_{.05}$	$\mathbf{.08}_{.04}$
NLL_{sum}	$\mathbf{.96}_{.07}$	$\underline{.10}_{.04}$
SE_{sum}	$.94_{.08}$	$\underline{.10}_{.05}$
SE_{max}	$\underline{.95}_{.07}$	$\mathbf{.08}_{.07}$
NLL_{max}	$.90_{.43}$	$.22_{.15}$
SE_{mean}	$.84_{.21}$	$.17_{.10}$
MTP	$.90_{.42}$	$.23_{.14}$
NLL_{avg}	$.77_{.23}$	$.21_{.07}$
LNTP	$.61_{.50}$	$.23_{.14}$
PPL	$.77_{.23}$	$.21_{.06}$

Table 3: Median (IQR as subscript) across 9 LLMs under the Extremes training group.

RQ1: Accuracy estimation under deployment-plausible defaults. We report results for two training groups chosen *a priori* to reflect plausible supervision strategies rather than tuned for best score: **Extremes** (GSM8K + OlympiadBench), which pairs an elementary and a competition-level benchmark to span the widest difficulty range, and **Intermediate** (MATH + SciBench), which covers mid-range difficulty. Both use a random forest with class balancing, isotonic calibration, and the 10 entropy-distribution statistics as the feature subset. Table 2 and Figure 3 report cross-domain accuracy estimation quality.

Both training groups generalize out of domain for most LLMs, but Extremes is consistently stronger: seven of nine models achieve $\rho \geq 0.90$ with low error (AEE 0.03–0.12), while Intermediate yields systematically higher error and weaker ordering across all models (AEE 0.06–0.17). The signal is also model-dependent: PHI-3.5-MINI

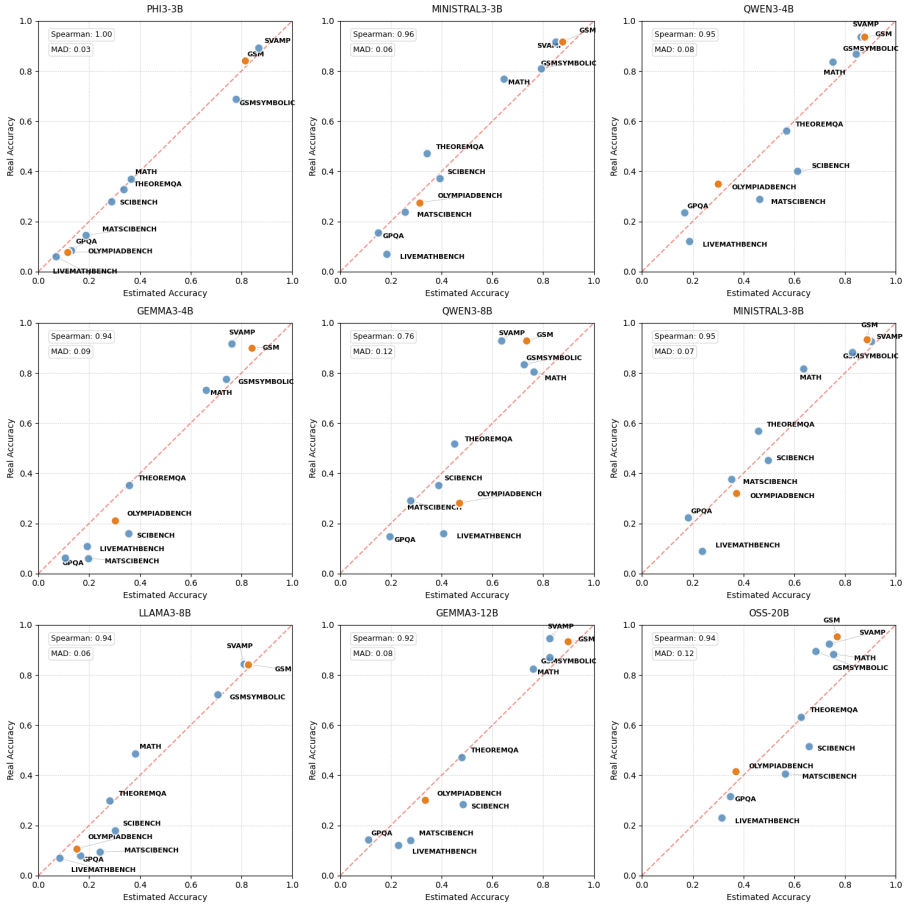

Figure 3: Accuracy estimations from a random-forest classifier trained exclusively on compact entropy-profile features on GSM and OlympiadBench. Both train benchmarks span the two extremes of difficulty.

achieves near-perfect ordering (AEE 0.03, $\rho = 1.00$), whereas QWEN3-8B exhibits weaker agreement in both settings (AEE 0.12–0.17, $\rho \approx 0.75$).

We also find that slice-level AEE is largely decoupled from per-instance AUROC (Pearson $R^2 < 0.06$): aggregate estimates stay accurate even on (LLM, benchmark) pairs where $\hat{P}(x)$ is a near-chance instance-level discriminator. See Appendix F. Additional training groups and estimator variants are reported in Appendix G.

RQ2: Comparison with single-metric baselines. Having established that the entropy-profile estimator generalizes well under plausible defaults, we ask whether this requires combining multiple features into a richer classifier, or whether a single calibrated UQ metric suffices. We compare the RQ1 estimator against the single-metric Platt-scaled baselines as described in Section 4, trained on the same Extremes group. Table 3 summarizes median performance and variability across the LLMs.

Three baselines are remarkably performant: NLL_{sum}, SE_{sum}, and SE_{max} achieve ρ and AEE in the same ballpark as the default estimator, with the default being slightly more consistent across LLMs. The remaining baselines degrade considerably—median ρ drops to 0.61–0.90 with large variability (IQR up to 0.50), and median AEE roughly doubles. This separation is consistent with the AUROC analysis in Section 3, where accumulation and extrema metrics were also the strongest individual discriminators of correctness. Overall, the narrow gap between the top baselines and the multi-feature estimator suggests that training composition and having accumulation or extrema metrics within the feature set is far more important than classifier architecture or feature dimensionality.

However, this near-parity holds only in the median: individual metrics exhibit high cross-model variance that the multi-feature estimator avoids. For instance, under the same Extremes training group, LNTP collapses to $\rho = 0.14$ on MINISTRAL3-3B and $\rho = 0.36$ on LLAMA-3.1-8B, while NLL_{max} drops to $\rho = 0.42$ on GPT-OSS (Table 10, Appendix H). The default estimator never falls below $\rho = 0.76$ on any model. The multi-feature profile thus acts as a robustness hedge: it sacri-

fices little at the median while preventing catastrophic failures on models where any single metric's entropy–correctness coupling breaks down. This is consistent with the decoupling between instance-level discrimination and slice-level calibration (Appendix F): the multi-feature classifier achieves more stable calibration across model families, even when individual metrics offer comparable discrimination on favorable models.

RQ3: Sensitivity to training composition. We quantify how accuracy estimation depends on the composition of the supervision set. Table 4 aggregates results over all training groups G of size $k \in \{1, 2, 3, 4\}$ and over classifier configurations, reporting median AEE and its IQR. Increasing k produces two consistent effects across all nine LLMs: median error decreases monotonically, and sensitivity to benchmark choice shrinks sharply. At k=1, median AEE ranges from 0.20 to 0.27 with high variability; by k=4, it improves to 0.06–0.16 with IQR compressed to 0.03–0.04. Benchmark choice is thus a first-order design decision at small k, but becomes less critical as supervision spans more tasks.

Difficulty balance explains much of the composition effect. To make this dependence interpretable, we summarize each training group G by its weighted average accuracy—the mean accuracy $A(D)$ over benchmarks $D \in G$ and LLMs, weighted by instance counts in the group—and relate it to held-out estimation quality. We perform this under the RQ1 configuration to control for architecture variability. Figure 4 reveals a U-shaped relationship: supervision sets that are too easy or too hard generalize worse, whereas groups with intermediate weighted accuracy (roughly 0.4–0.7) yield the lowest AEE. This is consistent with difficulty-diverse groups exposing the estimator to both low-entropy success patterns and high-entropy failure patterns, while homogeneous groups under-represent one regime and miscalibrate under domain shift. We analyze the best and worst benchmark combinations and a leave-one-out sensitivity study in Appendix I.

RQ4: Sensitivity to estimator design. We isolate estimator design effects by aggregating across all training groups and models. Table 5 reports main effects on median AEE and Spearman correlation across classifier architecture, calibration, balancing, and feature subsets.

Differences across all design axes are small: classifier choice and feature subset each shift median AEE by at most 0.03–0.04 points, calibration and balancing have near-zero effect, and Spearman remains stable at $\rho \approx 0.94$ throughout. Among the modest differences that do emerge, random forests with the 10D entropy-distribution statistics perform best—consistent with the plausible defaults chosen in RQ1—while MLPs show slight degradation, likely due to overfitting under limited supervision. The 17D profile performs slightly worse than 10D, suggesting the additional white-box UQ features introduce noise rather than complementary signal in this setting. Reducing to 3D or even 1D (SE_{sum} alone) remains competitive, aligning with the RQ2 finding that accumulation and extrema metrics already capture most of the useful signal.

Overall, estimator design is a second-order concern: the best and worst classifier–feature combinations differ by only 0.04 in median AEE, whereas training composition (RQ3) shifts median AEE by up to 0.10–0.15. This confirms the pattern suggested in RQ2, where simple Platt-scaled baselines under the same Extremes training group approached the multi-feature estimator—the choice of supervision data dominates both feature dimensionality and classifier complexity.

Practical takeaways. The most important design

	Training Benchmarks (k)			
Model	**1**	**2**	**3**	**4**
PHI-3.5-MINI (3.6B)	$.20_{.11}$	$\mathbf{.09}_{.07}$	$\mathbf{.07}_{.04}$	$\mathbf{.06}_{.03}$
MINISTRAL3 (3B)	$.24_{.09}$	$.12_{.07}$	$.10_{.04}$	$.09_{.03}$
MINISTRAL3 (8B)	$.20_{.10}$	$.13_{.07}$	$.11_{.05}$	$.10_{.04}$
QWEN3 (4B)	$.21_{.08}$	$.12_{.06}$	$.11_{.04}$	$.09_{.03}$
QWEN3 (8B)	$.27_{.06}$	$.21_{.05}$	$.18_{.04}$	$.16_{.04}$
GEMMA3 (4B)	$.26_{.12}$	$.18_{.08}$	$.13_{.05}$	$.12_{.04}$
GEMMA3 (12B)	$.23_{.10}$	$.15_{.07}$	$.12_{.05}$	$.10_{.04}$
LLAMA-3.1 (8B)	$.23_{.09}$	$.14_{.07}$	$.11_{.05}$	$.11_{.03}$
GPT-OSS (20B)	$.22_{.05}$	$.19_{.06}$	$.17_{.04}$	$.16_{.04}$

Table 4: Median AEE vs. k benchmarks (IQR subscripts), aggregated over all groups and architectures.

Factor	**Setting**	**AEE**	ρ
Classifier	RF	$\mathbf{.11}_{.07}$	$.94_{.06}$
	LR	$.13_{.07}$	$.94_{.06}$
	MLP	$.15_{.08}$	$.94_{.06}$
Calibration	Y / N	$.13_{.07}$ / $.13_{.08}$	$.94_{.06}$ / $.94_{.06}$
Balancing	Y / N	$.13_{.07}$ / $.13_{.08}$	$.94_{.06}$ / $.94_{.06}$
Features	17D	$.14_{.08}$	$.94_{.06}$
	10D	$\mathbf{.11}_{.06}$	$\mathbf{.95}_{.05}$
	3D	$.13_{.07}$	$.94_{.07}$
	1D	$.13_{.07}$	$.94_{.06}$

Table 5: Ablation main effects: median AEE and Spearman ρ (IQR as subscripts), aggregated across all training groups and models. RF=Random Forest, LR=Logistic Regression, MLP=Multi Layer Perceptron.

choice, above predictor or feature selection, is the diversity of the supervision set. Training groups that span difficulty consistently outperform homogeneous groups, which underrepresent either high-entropy failure patterns (easy-only) or low-entropy success patterns (hard-only). With this choice made well, entropy profiles from decoding logs provide a useful signal for domain-level accuracy estimation: with just two benchmarks, we often generalize to eight unseen domains, and in the best case (PHI-3.5-MINI) we observe near-perfect alignment with ground-truth accuracies. Reliability is model-dependent, however, so the method should be validated on the target model before deployment. For seven of nine LLMs, the Extremes configuration yields $\rho \geq 0.90$ and AEE ≤ 0.12. The two partial exceptions are instructive. QWEN3-8B exhibits weaker entropy–correctness coupling than its smaller QWEN3-4B counterpart: several statistics show near-chance AUROC on benchmarks where 4B achieves strong separability (Table 8, Appendix C); the method should not be assumed to improve with model scale. GPT-OSS presents a different pattern: strong ranking ($\rho = 0.90$–0.96) but systematic absolute offset (AEE $= 0.15$–0.20), recoverable with a simple affine correction. Overall, these edge cases reinforce our recommendation to validate on the targey model, while confirming that the approach is effective across families.

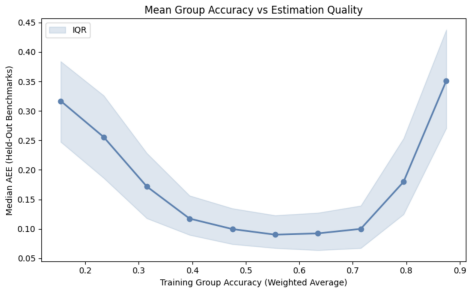

Figure 4: Relationship between training group difficulty and estimation quality aggregated across all nine LLMs. Training groups with intermediate weighted accuracy (0.4–0.6) achieve optimal performance, while difficulty-homogeneous groups at either extreme degrade generalization. Shaded region indicates IQR.

6 Related Work

Performance estimation from internal states. LLM internal states encode correctness information, exploited for instance-level hallucination detection (Azaria and Mitchell, 2023; Chen et al., 2024; Duan et al., 2024; Ahdritz et al., 2024), though outputs and internal states can disagree (Liu et al., 2023). White-box approaches use token-probability metrics (Malinin and Gales, 2021; Fadeeva et al., 2024; Kuhn et al., 2023); sampling-based methods assess consistency (Manakul et al., 2023) at the cost of multiple generations. Unlike prior work targeting per-instance correctness, we estimate *domain-level accuracy* from standard inference logs.

Supervised approaches using internal signals. Prior work trains classifiers on hidden states (Azaria and Mitchell, 2023; Chen et al., 2024; Zhang et al., 2025a) or ensembles uncertainty signals (Chen and Mueller, 2023; Verga et al., 2024; Bouchard and Chauhan, 2025a). Recent single-pass methods use top-k log-probabilities (Moslonka et al., 2025) or sequential entropy modeling (Vathul et al., 2025) for instance-level detection. We likewise use single-pass entropy but shift the objective to continuous domain-level accuracy, which is not necessarily aligned (Appendix F).

Entropy-Lens. Our focus on entropy signals is motivated by Entropy-Lens (Ali et al., 2025), which uses Shannon entropy traces across transformer layers to classify model family, task type, and correctness. While effective, full residual-stream analysis is costly and yields high-dimensional features that may generalize poorly OOD. In contrast, we train on compact final-layer entropy summaries, requiring only top-k log-probabilities available from standard serving APIs.

7 Conclusion

We studied whether decoding entropy traces can support continuous accuracy monitoring under domain shift in STEM reasoning. Across an exhaustive sweep over ten benchmarks, nine LLMs (3B–20B), and $>$160k estimator configurations, entropy-based estimates often track held-out accuracies closely and preserve domain rankings, though reliability is model-dependent. The dominant factor is supervision composition: difficulty-spanning training sets generalize substantially better than homogeneous ones. Our results provide evidence that entropy profiles are a useful signal for continuous monitoring; the resulting per-slice rankings could also guide data acquisition, though validating this downstream loop and extending beyond verifiable STEM tasks remain future work.

8 Limitations

Controlled task: verifiable reasoning. Our experiments focus on ten benchmarks with well-defined correctness criteria and zero-shot prompting. This enables large-scale, exhaustive train/test sweeps, but it also limits external validity: open-ended tasks (e.g., creative writing, dialogue, summarization) do not admit a single gold answer. The methodology itself is not tied to binary correctness, though: the classifier's supervision label can be substituted for any per-instance quality signal (e.g., factuality scores from claim-level verifiers, faithfulness judgments, etc.), and the aggregation step (Eq. 1) is unchanged. Whether entropy features remain predictive under these noisier labels is left to future work.

Controlled domains: math and science. Our benchmarks all fall within math and science. Whether the method generalizes to other varied domains — law, economics, medicine — where entropy patterns under correctness may differ, remains untested. The main concern regarding this is that production slices are rarely single-domain to begin with. One possible path forward is to use the estimator as a downstream component, with an upstream router dispatching inputs to domain-specialized sentinels trained on narrower domains.

Sensitivity to decoding and formatting. Entropy traces depend on decoding choices (temperature, max length, stop criteria) and on the model's tendency to produce longer chain-of-thought or verbose explanations. Changes in prompting (e.g., instructing shorter solutions), answer formatting, or post-processing rules can shift entropy distributions without reflecting true capability changes.

Actionability beyond ranking. While Spearman ρ indicates that we can often rank domains by difficulty, absolute accuracy estimation error (AEE) remains non-trivial for some models, and miscalibration can persist even under large supervision (e.g., strong ranking but systematic offset). In operational settings, this suggests using the method primarily for *prioritization* (which slices to inspect or label) and coupling it with targeted evaluation before high-stakes interventions.

References

Gustaf Ahdritz, Tian Qin, Nikhil Vyas, Boaz Barak, and Benjamin L. Edelman. 2024. Distinguishing the knowable from the unknowable with language models. *arXiv preprint arXiv:2402.03563*.

Riccardo Ali, Francesco Caso, Christopher Irwin, and Pietro Liò. 2025. Entropy-lens: The information signature of transformer computations. *Preprint*, arXiv:2502.16570. Preprint. Under review.

Anonymous. 2025. Livemathbench: A contamination-resistant dynamic math reasoning benchmark. *arXiv preprint arXiv:2505.15340*.

Amos Azaria and Tom Mitchell. 2023. The internal state of an LLM knows when it's lying. *arXiv preprint arXiv:2304.13734*.

Dylan Bouchard and Mohit Singh Chauhan. 2025a. Uncertainty quantification for language models: A suite of black-box, white-box, LLM judge, and ensemble scorers. *Preprint*, arXiv:2504.19254.

Dylan Bouchard and Mohit Singh Chauhan. 2025b. UQLM: A Python package for uncertainty quantification in large language models. *Preprint*, arXiv:2507.06196.

Chao Chen, Kai Liu, Ze Chen, Yi Gu, Yue Wu, Mingyuan Tao, Zhihang Fu, and Jieping Ye. 2024. INSIDE: LLMs' internal states retain the power of hallucination detection. *arXiv preprint arXiv:2402.03744*.

Jiuhai Chen and Jonas Mueller. 2023. Quantifying uncertainty in answers from any language model and enhancing their trustworthiness. *arXiv preprint arXiv:2308.16175*.

Wenhu Chen and 1 others. 2023. Theoremqa: A theorem-driven question answering dataset. In *Proc. EMNLP*.

Karl Cobbe and 1 others. 2021. Training verifiers to solve math word problems. In *arXiv preprint arXiv:2110.14168*.

Hanyu Duan, Yi Yang, and Kar Yan Tam. 2024. Do LLMs know about hallucination? an empirical investigation of LLM's hidden states. *arXiv preprint arXiv:2402.09733*.

Ekaterina Fadeeva, Aleksandr Rubashevskii, Artem Shelmanov, Sergey Petrakov, Haonan Li, Hamdy Mubarak, Evgenii Tsymbalov, Gleb Kuzmin, Alexander Panchenko, Timothy Baldwin, Preslav Nakov, and Maxim Panov. 2024. Fact-checking the output of large language models via token-level uncertainty quantification. *arXiv preprint arXiv:2403.04696*.

Google DeepMind. 2025. Gemma 3: Multimodal open models built from gemini technology. Technical report, Google.

Chaoqun He and 1 others. 2024. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. In *Proc. ACL*.

Dan Hendrycks and 1 others. 2021. Measuring mathematical problem solving with the math dataset. In *Proc. NeurIPS*, pages 1–15.

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, and 17 others. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. *arXiv preprint arXiv:2302.09664*.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th Symposium on Operating Systems Principles (SOSP)*.

Kevin Liu, Stephen Casper, Dylan Hadfield-Menell, and Jacob Andreas. 2023. Cognitive dissonance: Why do language model outputs disagree with internal representations of truthfulness? *arXiv preprint arXiv:2312.03729*.

Andrey Malinin and Mark Gales. 2021. Uncertainty estimation in autoregressive structured prediction. In *International Conference on Learning Representations*.

Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. 2023. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *arXiv preprint arXiv:2303.08896*.

Meta AI. 2024. The llama 3 herd of models. Technical report, Meta.

Microsoft. 2024. Phi-3.5-mini technical report. Technical report, Microsoft Research.

Iman Mirzadeh and 1 others. 2025. Gsm-symbolic: Understanding the limitations of mathematical reasoning in large language models. In *Proc. ICLR*.

Mistral AI. 2025. Ministral 3: Efficient frontier models for local reasoning. Release Announcement.

Charles Moslonka, Hicham Randrianarivo, Arthur Garnier, and Emmanuel Malherbe. 2025. Learned hallucination detection in black-box llms using token-level entropy production rate. *Preprint*, arXiv:2509.04492.

OpenAI. 2025. Gpt-oss model card: Transparent frontier models. Technical report, OpenAI.

Arkil Patel and 1 others. 2021. Are nlp models really able to solve simple math word problems? In *Proc. NAACL*, pages 2080–2094.

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

Qwen Team. 2025. Qwen 3: Innovative multimodal llm and reasoning models. Technical report, Alibaba Group.

David Rein and 1 others. 2023. Gpqa: A graduate-level google-proof q&a benchmark. *arXiv preprint arXiv:2311.12022*.

Aneesh Vathul, Daniel Lee, Sheryl Chen, and Arthi Tasmia. 2025. Shed-hd: A shannon entropy distribution framework for lightweight hallucination detection on edge devices. *Preprint*, arXiv:2503.18242.

Pat Verga, Sebastian Hofstatter, Sophia Althammer, Yixuan Su, Aleksandra Piktus, Arkady Arkhangorodsky, Minjie Xu, Naomi White, and Patrick Lewis. 2024. Replacing judges with juries: Evaluating LLM generations with a panel of diverse models. *arXiv preprint arXiv:2404.18796*.

Xiaoxuan Wang and 1 others. 2023. Scibench: Evaluating college-level scientific problem solving. *arXiv preprint arXiv:2307.10635*.

xAI. 2025. Grok 4.1 fast: Frontier tool-use and reasoning agents. XAI Blog.

Anqi Zhang, Yulin Chen, Jane Pan, Chen Zhao, Aurojit Panda, Jinyang Li, and He He. 2025a. Reasoning models know when they're right: Probing hidden states for self-verification. In *Conference on Language Modeling (COLM)*.

Anqi Zhang and 1 others. 2025b. Matscibench: Benchmarking the reasoning ability of large language models in materials science. *arXiv preprint arXiv:2510.12171*.

A Baseline White Box UQ Metrics

We compare against nine standard uncertainty quantification baselines computed exclusively from the output token logprobs of a single forward pass. Most can be found in the uqlm library (Bouchard and Chauhan, 2025b). These constitute our most direct comparison: cheap signals available from any standard inference call, suitable for continuous performance monitoring.

Shannon Entropy (SE). Following Malinin and Gales (2021); Manakul et al. (2023), we compute token-level entropy over vocabulary $\mathcal{V}$:

$$\mathrm{SE}(y_i) = -\sum_{v \in \mathcal{V}} P(v \mid x, y_{<i}) \log P(v \mid x, y_{<i}) \tag{2}$$

In practice, the sum is truncated to the top $k = 20$ tokens. We aggregate as: $\mathrm{SE}_{\text{avg}}(y) = \frac{1}{L}\sum_{i=1}^{L} \mathrm{SE}(y_i)$, $\mathrm{SE}_{\max}(y) = \max_i \mathrm{SE}(y_i)$, and $\mathrm{SE}_{\text{sum}}(y) = \sum_{i=1}^{L} \mathrm{SE}(y_i)$ (Shannon Entropy Accumulation, abbreviated SEA).

Stat	MATH	GSM8K	OLYMP.
Mean	+0.002	+0.000	+0.002
Std	+0.005	−0.001	+0.003
Max	+0.007	−0.000	−0.007
Q10	+0.000	+0.001	−0.000
Q25	−0.001	+0.002	+0.002
Q50	+0.004	+0.002	+0.000
Q75	+0.002	−0.001	−0.001
Q90	+0.004	+0.002	−0.000
Skew	−0.016	−0.002	−0.021
Kurt	−0.020	−0.003	−0.032
SEA	+0.001	−0.000	+0.000

Table 6: AUROC change (full vocabulary − top-20) for Phi-3.5-Mini on MATH, GSM8K, and OlympiadBench.

Negative Log-Likelihood (NLL). Manakul et al. (2023) use $\text{NLL}(y_i) = -\log P(y_i \mid x, y_{<i})$, aggregated as: $\text{NLL}_{\text{avg}}(y) = \frac{1}{L}\sum_{i=1}^{L}\text{NLL}(y_i)$, $\text{NLL}_{\text{max}}(y) = \max_i \text{NLL}(y_i)$, and $\text{NLL}_{\text{sum}}(y) = \sum_{i=1}^{L}\text{NLL}(y_i)$.

Length-Normalized Token Probability (LNTP). Malinin and Gales (2021) propose the geometric mean of token probabilities:

$$\text{LNTP}(y) = \left(\prod_{i=1}^{L} P(y_i \mid x, y_{<i})\right)^{1/L} \quad (3)$$

Minimum Token Probability (MTP). Manakul et al. (2023) identify the weakest link: $\text{MTP}(y) = \min_i P(y_i \mid x, y_{<i})$.

Perplexity (PPL). Fadeeva et al. (2024) propose standard perplexity: $\text{PPL}(y) = \exp(\text{NLL}_{\text{avg}}(y))$.

B Effect of logprob truncation on predictive power

Our entropy profiles use the top-20 logprobs from inference. To verify that this truncation does not distort predictive power, we recomputed every Phi-3.5-mini profile from the *full-vocabulary* log-softmax. The resulting AUROC deltas (Table 6) are negligible: the typical shift is on the order of 0.002, and every location and scale statistic moves by less than 0.01. The largest deltas concentrate in skewness and kurtosis on the harder benchmarks (≈ 0.03 on MATH and OlympiadBench, ≈ 0.003 on GSM8K), yet favor the $k = 20$ truncation and remain small on the AUROC scale.

C Additional Entropy Profile Features Results

Table 8 provides AUROC results for the remaining five models, supplementing the diagnostic study in Sec. 3. We observe significant heterogeneity across families and sizes. QWEN-3 4B achieves exceptionally high separability on MATH (Max: 0.9196), while its larger counterpart, QWEN-3 8B, shows a performance inversion with near-chance AUROCs on several statistics, even on equal benchmarks. This showcases that entropy profiles and their discriminative potential vary considerably between models of even the same family. Consistent with our main results, lower-tail quantiles (Q10, Q25) are most predictive for difficult reasoning tasks like MATH, while central tendency measures (Mean, Std Dev) perform better on elementary tasks like GSM8K. Furthermore, the GEMMA-3 family exhibits family-specific brittleness in higher-order moments (Skewness, Kurtosis), which collapse toward chance performance on difficult benchmarks.

Results provide supporting evidence that entropy signals carry a strong correctness signal across families, where the discriminatory potential of individual statistics is highly model- and domain-dependent.

D Feature Sensitivity Analysis

We test whether per-statistic AUROC is robust to decoding choices. Holding model (Phi-3.5-Mini), benchmark (500 MATH-test problems), and item selection fixed, we run five seeds at each of $T \in \{0.3, 0.5, 0.7, 1.0\}$; at $T = 0.5$ the seed-42 run reuses the paper's original generations. Table 7 reports min–max AUROC ranges across seeds.

Seed-to-seed spreads are typically 0.02–0.03 AUROC, with the leading summaries (Q10, Q25, SEA, NLL_{sum}) tightening to ≤ 0.013 in several cells (e.g., SEA at T=1.0: 0.812–0.815); the widest spreads ($\sim$ 0.07–0.10, for Std and Max at T=0.5) still sit below the leaders. Temperature induces a structured shift: bulk and average-likelihood statistics (Mean, Std, Max, NLL_{avg}, LNTP, PPL) drift monotonically upward—every T=1.0 seed strictly exceeds every T=0.3 seed—while lower-tail quantiles (Q10, Q25) stay within 0.78–0.84 across all twenty runs and accumulation metrics (SEA, NLL_{sum}) drift only mildly ($\sim$ 0.02–0.04).

All in all, we consider that single-seed AUROC estimates at T=0.5 are a faithful proxy for the underlying signal: seed noise is small, temperature reshapes individual statistics in predictable ways without reordering them, and the feature choices motivating our compact entropy profile remain the right ones across the decoding regimes a practitioner is likely to use.

Stat	$T = 0.3$	$T = 0.5$	$T = 0.7$	$T = 1.0$
Mean	.657–.702	.657–.728	.717–.746	.724–.748
Std	.650–.694	.668–.764	.742–.762	.752–.773
Max	.706–.717	.723–.795	.770–.790	.767–.799
Q10	.812–.834	.806–.830	.829–.837	.801–.831
Q25	.795–.820	.794–.814	.810–.823	.783–.808
Q50	.715–.732	.714–.744	.750–.762	.733–.736
Q75	.653–.693	.659–.697	.690–.719	.697–.712
Q90	.664–.696	.646–.713	.694–.731	.715–.745
Skew	.586–.647	.582–.609	.603–.647	.603–.628
Kurt	.565–.628	.557–.578	.579–.618	.579–.604
SEA	.787–.816	.805–.826	.803–.839	.812–.815
NLL_{avg}	.629–.660	.625–.698	.688–.716	.713–.737
NLL_{max}	.664–.693	.669–.698	.671–.708	.679–.705
NLL_{sum}	.769–.787	.774–.817	.785–.830	.798–.813
LNTP	.629–.660	.625–.698	.688–.716	.713–.737
MTP	.664–.693	.669–.698	.671–.708	.679–.705
PPL	.629–.660	.625–.698	.688–.716	.713–.737

Table 7: Per-statistic AUROC ranges (min–max across seeds $s \in \{42, 43, 44, 45, 46\}$) for Phi-3.5-Mini on 500 MATH-test problems at four sampling temperatures. At $T = 0.5$ the seed-42 run uses the paper's original generations rather than a regenerated sample. Skew, Kurt, LNTP, and MTP report $1 - \text{AUROC}$, following the convention of Table 1.

E Additional details for Reproducibility

To ensure reproducibility across the ten STEM reasoning benchmarks and nine LLMs, we provide the specific prompts used for model generation and external validation. All benchmarks utilized zero-shot chain-of-thought (CoT) prompting.

E.1 Hardware and Compute

All experiments were run on a single NVIDIA A6000 GPU (48GB VRAM). Total inference time was approximately 78 hours across all nine LLMs and ten benchmarks.

E.2 Model Generation Prompt

We used the instruct-tunes of every LLM version ran through vLLM (Kwon et al., 2023), utilizing a single user prompt without a system message to maintain consistency across open and closed-source families:

```
Solve the following problem step by step,
then provide the final numerical answer.

Question: {question}

Solution:
```

E.3 Evaluation and Verification Prompt

Ground-truth correctness labels were produced using GROK-4.1-FAST-REASONING (xAI, 2025) via the LiteLLM API as an external validator. The validator receives the original question, the model's response, and the reference ground-truth answer, then outputs a structured binary decision using a Pydantic response schema (`class Response(BaseModel): success: bool`).

The system prompt instructs the validator to determine answer equivalence under benchmark-specific criteria (e.g., symbolic or numeric equivalence):

```
System: Task: Determine if the Response
corresponds to the Correct Answer for
the Question, based on the given Correct
Answer text. Answer ONLY with the exact
format: {"success": True} or {"success":
False}
```

The user prompt provides the evaluation context:

```
User: # Question
{question}

# Response
{cleaned_response}

# Correct Answer
{correct_answer}
```

Responses are preprocessed by removing special tokens (e.g., `<|end|>`, `<|endoftext|>`) before validation. The structured Pydantic output ensures consistent binary classification across all 385 training configurations and benchmark evaluations.

E.4 Classifier Hyperparameter Grid

All hyperparameters are selected via a 5 fold cross-validated grid search on the training group, optimizing ROC-AUC. For the logistic regression, we don't train for any hyperparameter. For random forests (100 estimators), we search max depth $\in \{3, 5, 10\}$ and min samples split $\in \{2, 5, 10\}$. For MLPs (ReLU activations, early stopping, $\alpha = 0.001$), we search hidden layer sizes $\in \{(5), (8), (10), (15), (20), (8, 4), (10, 5), (15, 8)\}$.

E.5 Class Balancing Techniques

We evaluate each classifier with and without class balancing using scikit-learn's built-in mechanisms. Logistic regression uses `class_weight="balanced"`, reweighting the loss inversely proportional to class frequencies. Random forests use `class_weight="balanced_subsample"`, reweighting within each bootstrap sample. For MLPs, which lack native class weighting, we apply random oversampling via `RandomOverSampler` from imbalanced-learn [3] prior to training.

F Decoupling between slice AEE and per-instance AUROC

Good domain-level estimates do not require good instance-level predictions. AEE captures the bias

[3] https://imbalanced-learn.org/stable/

Statistic	MATH	GSM8K	OLYMP.	Statistic	MATH	GSM8K	OLYMP.
	MINISTRAL-3 3B				QWEN-3 8B		
Mean	0.7512	0.7823	0.8225	Mean	0.5698	0.7726	0.7935
Std Dev	0.7648	0.7873	0.8077	Std Dev	0.5966	0.7776	0.8104
Max	0.7856	0.7771	0.7764	Max	0.5392	0.7117	0.7323
Q10	0.7887	0.7741	0.8099	Q10	**0.6484**	0.7509	**0.8236**
Q25	0.7749	0.7679	0.8230	Q25	0.5926	0.7614	0.8128
Q50	0.7558	0.7726	0.8328	Q50	0.5279	0.7485	0.7820
Q75	0.7332	0.7775	0.8259	Q75	0.5593	0.7701	0.7829
Q90	0.7504	0.7754	0.8130	Q90	0.5914	0.7802	0.8041
Skewness	0.7145	0.7457	0.8229	Skewness	0.5520	0.7633	0.7699
Kurtosis	0.7113	0.7368	0.8199	Kurtosis	0.5617	0.7652	0.7698
SEA	**0.8612**	**0.8145**	**0.8700**	SEA	0.6014	**0.8032**	0.7950
NLL_{avg}	0.6168	0.7256	0.7864	NLL_{avg}	0.5689	0.7701	0.7907
NLL_{max}	0.3681	0.4432	0.4345	NLL_{max}	0.5125	0.6587	0.6028
NLL_{sum}	0.8571	0.8009	0.8661	NLL_{sum}	0.5993	0.8012	0.7923
LNTP	0.6168	0.7256	0.7864	LNTP	0.5689	0.7701	0.7907
MTP	0.3681	0.4432	0.4345	MTP	0.5125	0.6587	0.6028
PPL	0.6168	0.7256	0.7864	PPL	0.5689	0.7701	0.7907
	QWEN-3 4B				LLAMA-3 8B		
Mean	0.8759	0.8093	0.8679	Mean	0.5557	0.7319	0.5263
Std Dev	0.9011	0.8100	0.8849	Std Dev	0.6431	0.7602	0.5875
Max	0.9196	0.8144	0.8472	Max	0.8404	0.7520	0.7465
Q10	0.8607	0.7452	0.8705	Q10	0.6763	0.7542	0.5753
Q25	0.8373	0.7605	0.8631	Q25	0.6614	0.7538	0.5861
Q50	0.8133	0.7937	0.8523	Q50	0.5813	0.7097	0.5372
Q75	0.8299	0.8148	0.8571	Q75	0.5091	0.7110	0.4946
Q90	0.8712	0.8003	0.8732	Q90	0.5784	0.7470	0.5392
Skewness	0.8185	0.7649	0.8466	Skewness	0.4448	0.6457	0.4539
Kurtosis	0.8153	0.7485	0.8483	Kurtosis	0.4494	0.6378	0.4578
SEA	**0.9327**	**0.8193**	0.8979	SEA	**0.8861**	**0.7877**	**0.8555**
NLL_{avg}	0.8746	0.7869	0.8683	NLL_{avg}	0.5400	0.7192	0.5127
NLL_{max}	0.8526	0.7348	0.7899	NLL_{max}	0.7957	0.6758	0.6974
NLL_{sum}	0.9325	0.8051	**0.8981**	NLL_{sum}	0.8823	0.7799	0.8523
LNTP	0.8746	0.7869	0.8683	LNTP	0.5400	0.7192	0.5127
MTP	0.8526	0.7348	0.7899	MTP	0.7957	0.6758	0.6974
PPL	0.8746	0.7869	0.8683	PPL	0.5400	0.7192	0.5127
	GEMMA-3 4B				GEMMA-3 12B		
Mean	0.6456	0.7118	0.8223	Mean	0.5522	0.6899	0.7891
Std Dev	0.7130	0.7316	0.8473	Std Dev	0.6477	0.7097	0.8118
Max	0.8453	0.7267	0.8328	Max	0.8334	0.7403	0.8196
Q10	0.8013	0.7354	0.8907	Q10	0.8061	0.7608	0.8487
Q25	0.8099	0.7425	**0.8916**	Q25	0.8322	**0.7644**	**0.8536**
Q50	0.7720	0.7275	0.8754	Q50	0.7334	0.7312	0.8377
Q75	0.6560	0.6955	0.8299	Q75	0.5661	0.6891	0.7907
Q90	0.6080	0.6944	0.8023	Q90	0.5108	0.6656	0.7712
Skewness	0.4843	0.6241	0.7273	Skewness	0.4044	0.5871	0.7008
Kurtosis	0.4700	0.6087	0.7153	Kurtosis	0.3963	0.5674	0.6913
SEA	**0.8672**	0.7760	0.8768	SEA	**0.8880**	0.7416	0.8495
NLL_{avg}	0.6328	0.7059	0.8101	NLL_{avg}	0.5273	0.6828	0.7758
NLL_{max}	0.7513	0.6890	0.7246	NLL_{max}	0.8088	0.7191	0.7379
NLL_{sum}	0.8634	**0.7807**	0.8764	NLL_{sum}	0.8826	0.7377	0.8478
LNTP	0.6328	0.7059	0.8101	LNTP	0.5273	0.6828	0.7758
MTP	0.7513	0.6890	0.7246	MTP	0.8088	0.7191	0.7379
PPL	0.6328	0.7059	0.8101	PPL	0.5273	0.6828	0.7758

Table 8: AUROC for entropy profile summaries and UQ baselines across remaining models. Skewness, Kurtosis, LNTP, and MTP report $1 - \text{AUROC}$. Accumulation and extreme values are consistently the most discriminative metrics.

of $\hat{A}(D)$ against $A(D)$, while per-instance AUROC measures how well $\hat{P}(x)$ separates correct from incorrect generations. The averaging in Eq. 1 decouples the two: per-instance errors that balance across D cancel under averaging but still suppress AUROC. Aggregate estimation therefore depends on the *calibration* of $\hat{P}(x)$, not its discrimination.

Figure 5 showcases this. Each point is one (LLM, held-out benchmark) pair under the Extremes and Intermediate estimators. Pearson R^2 is 0.013 and 0.059: slice-level error stays low across the full AUROC range, including pairs where per-instance separation is near chance.

G Additional Classifier Configuration Results

We include two additional classifier configurations based on sensible *a priori* defaults, all utilizing class balancing and isotonic calibration. Each configuration represents a plausible practitioner choice without access to exhaustive hyperparameter search.

Cross-Domain Linear Estimator. We train a logistic regression on MATSCIBENCH (intermediate difficulty, materials science) paired with GSM-SYMBOLIC (elementary difficulty, symbolic perturbations of GSM8K). This configuration tests whether a simple linear model can generalize

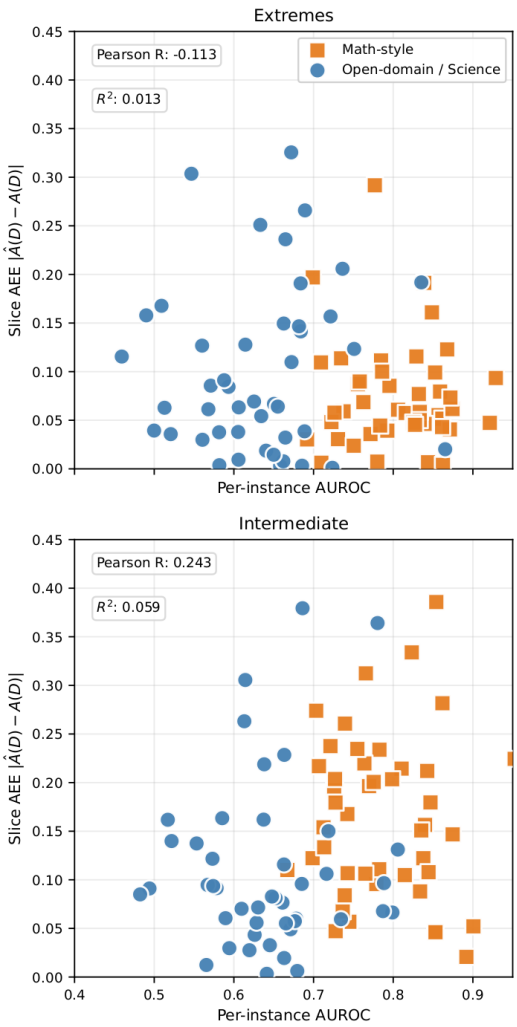

Figure 5: Slice AEE vs. per-instance AUROC under the Extremes and Intermediate estimators. Each point is one (LLM, held-out benchmark) pair. Pearson $R^2 = 0.013$ for Extremes and $R^2 = 0.059$ for Intermediate: slice-level error remains low across the full AUROC range.

across domains when trained on benchmarks that differ in both subject matter and problem format. The inclusion of GSM-Symbolic rather than standard GSM8K probes robustness to distribution shift within the elementary difficulty regime.

Science-Math Neural Estimator. We train an MLP on three benchmarks spanning scientific and mathematical reasoning: GPQA (graduate-level science), MATSCIBENCH (materials science), and GSM8K (elementary math). This configuration provides the neural estimator with cross-domain supervision while maintaining difficulty diversity, testing whether the additional representational capacity of MLPs benefits generalization when training data spans multiple reasoning types.

	Cross-Domain		Sci-Math	
Model	**AEE**	ρ	**AEE**	ρ
PHI-3.5-MINI (3B)	**0.06**	0.95	**0.07**	0.95
MINISTRAL3 (3B)	0.09	**0.98**	0.07	0.95
MINISTRAL3 (8B)	0.08	0.95	0.09	0.95
QWEN3 (4B)	0.08	0.96	0.10	**0.98**
QWEN3 (8B)	0.16	0.78	0.19	0.78
GEMMA3 (4B)	0.13	0.92	0.13	0.92
GEMMA3 (12B)	0.10	0.88	0.13	0.88
LLAMA-3.1 (8B)	0.09	0.95	0.08	0.94
GPT-OSS (20B)	0.16	0.94	0.15	0.93

Table 9: Cross-domain accuracy estimation for two classifier configurations: *Simple Cross-Domain* (logistic regression on MatSciBench + GSM-Symbolic) and *Flexible Sci-Math* (MLP on GPQA + MatSciBench + GSM8K). Both use class balancing and isotonic calibration.

Results. Table 9 reports cross-domain accuracy estimation for both configurations. Comparing against the main configurations (Table 2) reinforces two key findings from our main experiments.

Training composition dominates classifier choice. The *Extremes* configuration from the main paper remains the strongest overall, achieving the lowest AEE for 7/9 models with its difficulty-spanning design. Crucially, the *Cross-Domain* logistic regression—a simpler linear model trained on only two benchmarks—matches or outperforms the *Intermediate* random forest for most models. Results are similar on the Sci-Math MLP configuration –a more flexible model trained on varied domain and difficulty data–, suggesting that a well-composed training set matters more than classifier flexibility.

Calibration quality is model-dependent. Ranking stability (ρ) varies more across LLM families than across estimator configurations. PHI-3.5-MINI maintains $\rho \geq 0.95$ and AEE ≤ 0.07 across all four configurations (Extremes, Intermediate, Cross-Domain, Sci-Math), exhibiting consistently strong entropy–correctness coupling. In contrast, QWEN3-8B achieves only $\rho \approx 0.76$–0.78 regardless of training composition or classifier choice, indicating that its entropy signal is inherently less predictive of correctness. These results reinforce that practitioners should validate entropy-based estimation on their target model before deployment.

H Full Baseline Results

Full baseline results can be found in Table 10.

Model		Default	SE_{sum}	SE_{max}	SE_{mean}	NLL_{avg}	NLL_{max}	NLL_{sum}	LNTP	MTP	PPL
PHI-3.5-MINI (3B)	ρ	**1.00**	0.90	0.94	0.93	0.93	0.89	0.98	0.86	0.89	0.93
	AEE	**0.03**	0.08	0.06	0.12	0.13	0.12	0.08	0.13	0.12	0.13
QWEN3 (4B)	ρ	0.95	**0.98**	0.96	0.96	0.96	0.96	**0.98**	0.95	**0.98**	0.96
	AEE	0.08	0.10	**0.07**	0.13	0.13	0.11	0.10	0.11	0.12	0.13
QWEN3 (8B)	ρ	0.76	**0.92**	0.75	0.75	0.77	0.90	**0.92**	0.82	0.90	0.77
	AEE	**0.12**	0.18	0.25	0.22	0.22	0.27	0.19	0.25	0.27	0.22
MINISTRAL3 (3B)	ρ	0.96	**0.98**	0.95	0.96	0.73	0.56	0.96	0.14	0.61	0.73
	AEE	**0.06**	0.07	0.07	0.09	0.18	0.29	0.07	0.30	0.29	0.19
MINISTRAL3 (8B)	ρ	0.96	**0.98**	0.95	0.96	0.73	0.47	**0.98**	0.61	0.47	0.77
	AEE	**0.07**	0.08	0.08	0.10	0.19	0.27	0.08	0.20	0.27	0.19
GEMMA3 (4B)	ρ	**0.94**	0.88	0.89	0.84	0.87	0.92	0.88	0.38	0.92	0.87
	AEE	**0.09**	0.13	0.14	0.20	0.21	0.22	0.12	0.29	0.24	0.21
GEMMA3 (12B)	ρ	0.92	0.90	0.88	0.84	0.84	**0.93**	0.90	0.88	**0.93**	0.84
	AEE	**0.08**	0.11	0.13	0.20	0.21	0.17	0.11	0.18	0.18	0.21
LLAMA-3.1 (8B)	ρ	0.95	0.95	0.95	0.61	0.61	**0.99**	0.96	0.36	**0.99**	0.61
	AEE	0.07	**0.06**	0.08	0.28	0.28	0.13	**0.06**	0.29	0.14	0.28
GPT-OSS (20B)	ρ	0.90	0.94	**0.95**	0.75	0.47	0.42	0.94	0.60	0.42	0.50
	AEE	0.15	**0.10**	0.13	0.17	0.22	0.23	**0.10**	0.23	0.23	0.21

Table 10: Spearman ρ (higher is better) and AEE (lower is better) between UQ baselines and true domain accuracy on held-out benchmarks.

I Additional Training Sensibility Results

This appendix provides supplementary analyses of how training data composition affects accuracy estimation quality across all nine LLMs.

Best and Worst Benchmark Combinations. Table 11 identifies the highest- and lowest-performing benchmark combinations at each k, along with their weighted average group accuracy. The results directly corroborate the U-shaped relationship from Figure 4: best-performing groups fall within the intermediate accuracy regime (0.4–0.7).2

For $k \geq 2$, the best combinations pair elementary benchmarks (GSM8K) with difficult ones (OLYMPIADBENCH), achieving group accuracies squarely in the optimal range. This difficulty-spanning composition exposes the estimator to both low-entropy success patterns and high-entropy failure patterns, enabling robust transfer to unseen domains. Notably, the $k = 2$ combination (GSM + OLY) achieves median AEE of 0.087, matching the best $k = 4$ configuration despite using half the supervision. In contrast, all worst-performing combinations consist exclusively of difficult benchmarks, systematically underrepresenting low-entropy success signatures and causing the estimator to miscalibrate on easier test domains. This asymmetry reinforces a central finding: difficulty diversity in the supervision set is the main driver of better accuracy estimation.

Leave One Out Analysis To further stress-test the estimator under near-maximal supervision, we train on $k = 9$ benchmarks and evaluate on the single held-out domain. This setting provides the most favorable conditions for generalization while isolating per-benchmark estimation difficulty. Table 12 reports median AEE across all held-out benchmarks, the median held-out AEE (estimation error on the single excluded benchmark), and Spearman correlation ρ.

The leave-one-out results strongly replicate our main findings: even with nine training benchmarks, model-dependent variation in entropy–correctness coupling persists. PHI-3.5-MINI achieves the lowest overall error (AEE = 0.07, held-out AEE = 0.06) with excellent ranking ($\rho = 0.96$), while MINISTRAL3 and QWEN3-4B achieve near-perfect domain ordering ($\rho = 0.98$).

In contrast, QWEN3-8B and GPT-OSS continue to exhibit degraded performance even under maximal supervision. QWEN3-8B shows the weakest ranking agreement ($\rho = 0.82$) and highest held-out error variability (IQR = 0.17), suggesting that its entropy profiles provide fundamentally weaker correctness signal—a limitation that additional training data cannot fully overcome. Interestingly, GPT-OSS achieves strong ranking ($\rho = 0.96$) but exhibits systematically high absolute error (held-out AEE = 0.20), indicating well-calibrated *relative* estimates but poorly calibrated *absolute* probabilities. This pattern suggests that for GPT-OSS, our entropy profiles approach reliably rank domains by difficulty but consistently over- or under-estimate accuracy magnitudes.

The within-family inversions observed in the main experiments persist: QWEN3-4B substantially outperforms its larger QWEN3-8B counterpart (held-out AEE 0.08 vs. 0.18; ρ 0.98 vs.

k	Benchmarks	AEE	Acc.
	Best Performing		
1	OLY	$129_{.063}$	.260
2	GSM, OLY	$.087_{.047}$	.636
3	GPQA, GSM, OLY	$.087_{.046}$	.531
4	GPQA, GSM, MAT, OLY	$.086_{.048}$	.413
	Worst Performing		
1	LIVE	$.269_{.119}$	.114
2	GPQA, LIVE	$.259_{.093}$	.143
3	GPQA, LIVE, MAT	$.216_{.072}$	.197
4	GPQA, LIVE, MAT, SCI	$.191_{.064}$	.220

Table 11: Best and worst benchmark combinations for each k value. Median AEE with IQR shown as subscripts. Acc. reports weighted average group accuracy across LLMs. Abbreviations: OLY=OlympiadBench, GSM=GSM8K, MAT=MatSciBench, LIVE=LiveMathBench, SCI=SciBench.

Model	AEE	Held-out AEE	ρ
PHI-3.5-MINI (3B)	$\mathbf{.07}_{.03}$	$\mathbf{.06}_{.06}$	$.96_{.02}$
MINISTRAL3 (3B)	$.08_{.02}$	$.08_{.08}$	$\mathbf{.98}_{.04}$
MINISTRAL3 (8B)	$.10_{.02}$	$.08_{.09}$	$.95_{.00}$
QWEN3 (4B)	$.09_{.02}$	$.08_{.09}$	$\mathbf{.98}_{.00}$
QWEN3 (8B)	$.16_{.04}$	$.18_{.17}$	$.82_{.04}$
GEMMA3 (4B)	$.14_{.03}$	$.13_{.10}$	$.92_{.00}$
GEMMA3 (12B)	$.11_{.04}$	$.13_{.09}$	$.88_{.05}$
LLAMA-3.1 (8B)	$.10_{.01}$	$.10_{.08}$	$.94_{.01}$
GPT-OSS (20B)	$.15_{.02}$	$.20_{.15}$	$.96_{.02}$

Table 12: Median accuracy estimation error (AEE), held-out AEE (leave-one-out), and Spearman correlation (ρ) for $k = 9$ training benchmarks, with interquartile range (IQR) shown as subscripts.

0.82), and GEMMA3-4B and MINISTRAL3-3B shows comparable or exceeds GEMMA3-12B and MINISTRAL3-8B on ranking quality respectively. These results reinforce that practitioners should empirically validate the approach on their specific deployment model.

J License

Code for the paper is released under the MIT License.

K Use of AI-Assitants

AI assistants were used for spell checking and proofreading of the manuscript, and for assistance with minor utility scripts. No AI-generated text was included directly in the paper, and all research, methodology, and analysis were conducted by the authors.