CERTAINTY IS NOT JUST CORRECTNESS: RETHINKING TOKEN-LEVEL CERTAINTY IN LLM REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Token-level certainty is widely used as a proxy for correctness in LLM training and inference. However, the performance of certainty-based methods depends both on the information in certainty scores and on how those scores are used. We therefore directly assess certainty's predictive ability through controlled empirical evaluations across models and tasks. We distinguish two prediction targets: identifying questions a model is more likely to answer correctly and distinguishing correct from incorrect responses to the same question. In our experiments, certainty is generally better at identifying questions a model is likely to answer correctly than at distinguishing correct from incorrect responses to the same question. Certainty also varies systematically across token types and positions within words, reflecting local properties of words and text form. Information about question difficulty appears early in generation, while the weaker information about answer correctness is more concentrated near the end. These findings show that the information certainty provides for decisions depends on the prediction target, the model, the certainty metric, and which token positions in the response are included in aggregation. We further demonstrate the practical value of these findings for test-time compute. We allocate the number of responses using certainty early in generation and weight answer votes using certainty near the end of each response. Compared with a fixed-sampling majority-voting baseline, this approach increases overall accuracy from 78.71% to 79.54% while reducing the total cost in generated tokens by 82.4%.

1 INTRODUCTION

Assessing the correctness of large language model (LLM) reasoning is essential for deciding which outputs to trust and how to improve them. Token-level certainty provides a readily available signal: it summarizes the model's next-token distribution and can be aggregated over a generated response. Using information already produced during generation, certainty enables decisions about model outputs without ground-truth answers or a separately trained verifier (Kang et al., 2025; Fu et al., 2025). This accessibility makes certainty useful for both training and inference.

Across these uses, certainty serves as a proxy for correctness: higher scores are treated as evidence that an output is more likely to be correct. This premise connects response selection, which favors high-scoring candidates (Kang et al., 2025), to training with certainty rewards, which reinforces high-scoring responses (Zhao et al., 2025). It also motivates selecting actions based on their token probabilities in interactive tasks (Wang et al., 2024a) and using certainty to decide whether a reasoning path should continue or an answer is ready to return (Fu et al., 2025; Nogueira et al., 2025). Although these decisions use scores differently, they rely on the correctness information expected from certainty.

However, gains from using certainty are measured for the training or inference method as a whole. They therefore do not directly establish how reliably certainty predicts correctness or under which conditions that relationship holds. Previous work has directly evaluated response ranking (Taubenfeld et al., 2025; Wang et al., 2024a), compared aggregation rules (Kang et al., 2025; Fu et al., 2025), and controlled reasoning content (Kim & Kim, 2026). A systematic account connecting these measurements is still missing: what certainty predicts, how the information it provides depends on the way certainty is measured, and which uses that information supports. Without such an account, a method's gains offer little guidance on when the same signal can support other decisions. We address

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

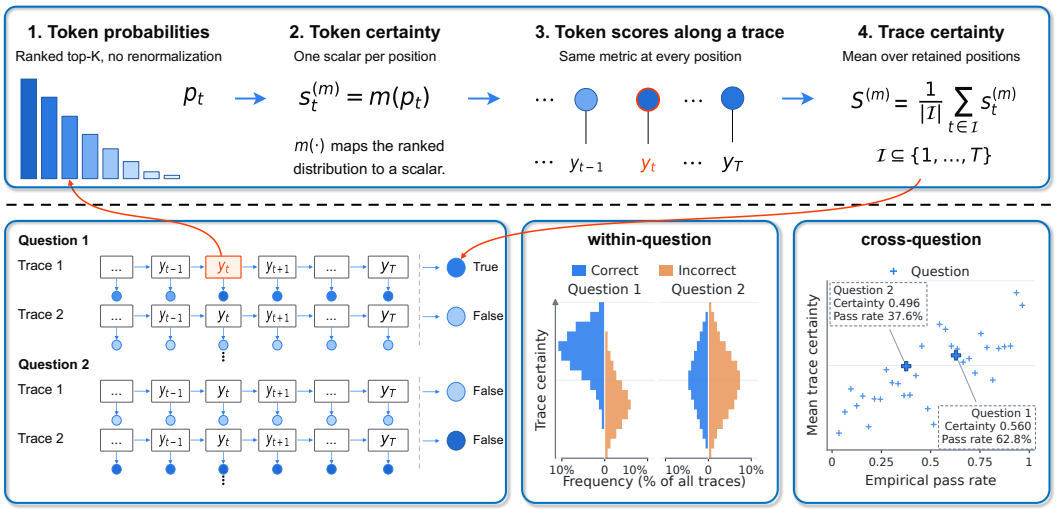


Figure 1: From token-level certainty to two prediction targets. A metric scores each next-token distribution; averaging scores over selected positions yields a response (or trace) score. Within-question comparisons test whether scores distinguish correct from incorrect responses to the same question. Cross-question comparisons test whether a question's mean response score reflects the proportion of its generated responses that answer the question correctly.

this gap through direct, controlled evaluation: we compare certainty scores against correctness labels on the same generated responses across models and tasks.

Our study covers five reasoning models, three mathematical and scientific benchmarks, and four certainty metrics. We distinguish identifying questions a model is more likely to answer correctly from identifying correct responses to the same question (Figure 1). With token scores averaged over complete responses, all four metrics are better at the former when results are averaged across benchmarks for each model. Information about question difficulty is already available early in generation, whereas the weaker information distinguishing correct and incorrect responses is more concentrated near the end. This difference motivates Target-Aligned Certainty Scaling (TACS). TACS uses scores from the beginning of a response to decide how many responses to generate for a question, and scores near the end to weight their votes for final answer selection. In our experiments, TACS reaches 79.54% accuracy at 17.6% of the generated-token cost of fixed 256-response majority voting, whose accuracy is 78.71%. Early budget allocation reduces token cost; weighting votes using certainty averaged over a final token window improves average accuracy, with gains that depend on the model. These findings show why certainty must be assessed in relation to the information a decision requires and when that information becomes available.

Our main contribution is a systematic empirical account of the information token-level certainty provides for reasoning. We find empirically that, across the models and tasks evaluated, certainty is generally better at identifying questions a model is likely to answer correctly than at distinguishing correct from incorrect responses to the same question. Its predictive ability varies with the model, task, and scoring choices. Certainty also reflects local properties of words and text form, and simply excluding selected tokens does not consistently improve prediction. Information about question difficulty is available early, whereas information about response correctness is more concentrated near the end. These findings suggest choosing which part of a response to score according to the information a decision requires. We examine the practical value of this principle through test-time compute experiments. TACS applies these principles to reduce generation costs, while gains from weighting response votes remain model-dependent.

2 Related Work

Token-level certainty. Token-level certainty summarizes the concentration of an LLM's next-token distribution at each generation position. We use a top-K version of **self-certainty** that averages

the negative log probabilities of the K most probable tokens (Kang et al., 2025; Fu et al., 2025). **Negative entropy** sums probability-weighted log probabilities (Kang et al., 2025; Kim & Kim, 2026); **Maximum probability** uses the highest next-token probability (Nogueira et al., 2025; Yang et al., 2025b). In our comparisons, we use its logarithm, maximum log probability (MaxLP). For ranked full-softmax probabilities $p_0 \geq \cdots \geq p_{K-1}$, we define these three metrics without renormalizing the top-K slice:

$$\begin{aligned} \text{Self-certainty } (\mathrm{SC}_K) &= -\frac{1}{K}\sum_{j=0}^{K-1}\log p_j, \\ \text{Top-}K\text{ Negative Entropy } (\mathrm{NegEnt}_K) &= \sum_{j=0}^{K-1} p_j \log p_j, \\ \text{Maximum Log Probability } (\mathrm{MaxLP}) &= \log p_0. \end{aligned} \tag{1}$$

From token certainty to trace scores. Aggregating token-level scores yields a score for a generated response, or trace. A typical approach is mean aggregation: token scores s_t are averaged over a selected set of positions $\mathcal{I}$:

$$S = \frac{1}{|\mathcal{I}|}\sum_{t\in\mathcal{I}} s_t. \tag{2}$$

Self-Certainty (Kang et al., 2025) averages scores over the entire response. DeepConf (Fu et al., 2025) also considers the mean over a tail segment and the sliding window with the lowest mean score. DEER (Yang et al., 2025b) averages per-position maximum probabilities over an induced trial answer. RENT (Prabhudesai et al., 2025) averages token-level negative entropy over response positions. These methods extend scoring from complete responses to local segments or trial answers.

From certainty scores to decisions. Model-internal probability signals guide certainty-based output selection, model training, and test-time compute control. For output selection, scores are used to choose candidate responses or actions (Kang et al., 2025; Wang et al., 2024a) and weight votes across responses (Kang et al., 2025; Taubenfeld et al., 2025). For model training, self-certainty (Zhao et al., 2025) and negative token entropy (Prabhudesai et al., 2025; Agarwal et al., 2025) serve as reinforcement learning rewards. Token entropy also supports selecting update positions, with policy-gradient updates restricted to high-entropy tokens in reinforcement learning with verifiable rewards (Wang et al., 2025). For test-time compute control, confidence or entropy signals guide reasoning-path termination and decisions about further reasoning (Fu et al., 2025; Nogueira et al., 2025; Yang et al., 2025b; Sharma & Chopra, 2025), as well as adjustments to the number of sampled responses (Kim et al., 2026). Using certainty to select, reinforce, or retain responses assumes that it helps distinguish correct from incorrect outputs. However, we still lack a systematic understanding of how well these signals distinguish correct from incorrect outputs across different models, tasks, choices of scoring metric, and the regions of a response used for scoring.

3 Evaluation Framework

The performance of certainty-based methods depends on both certainty scores and their use; performance alone does not establish how well the scores distinguish correct from incorrect responses. For example, consider a typical certainty application: weighted answer voting, which sums certainty scores for each answer and selects the answer with the largest total. Suppose four of five responses to a question give the same incorrect answer and one gives the correct answer. Even if the certainty score is 2 for the correct response and 1 for every incorrect response, yielding perfect correctness ranking, the incorrect answer still wins 4 to 2. Any permutation of these scores leaves the winner unchanged. Thus, the same voting outcome can occur with very different ranking quality; voting accuracy alone cannot evaluate certainty's discriminative ability. To measure this ability directly, we establish a controlled evaluation framework that compares scoring choices on the same generated responses and correctness labels. We separately measure how well scores distinguish correct from incorrect responses to the same question and how well they distinguish questions with different empirical accuracies for a fixed model. This distinction lets us identify what certainty predicts and examine how its discriminative ability varies with the metric, scoring position, model, and task.

3.1 Evaluation Objectives

When using certainty to distinguish correct from incorrect responses, we must also ask whether scores distinguish questions or responses to the same question. Consider two questions with ten responses each. The first has nine correct responses, and all ten responses score one; the second has one correct response, and all ten responses score zero. The AUROC is 0.9 when all twenty responses are combined, but 0.5 for each question separately. The score separates questions but cannot select correct responses within either. We therefore distinguish discrimination among responses to the same question from discrimination of relative question difficulty for a fixed model.

For question q with n_q responses, let S_{qi} denote certainty and $y_{qi} \in \{0, 1\}$ correctness. Define

$$\bar{S}_q = \frac{1}{n_q} \sum_{i=1}^{n_q} S_{qi}, \qquad \hat{r}_q = \frac{1}{n_q} \sum_{i=1}^{n_q} y_{qi}. \tag{3}$$

Here $\bar{S}_q$ is mean certainty and $\hat{r}_q$ is empirical accuracy. For each question q, let i^+ and i^- be indices drawn independently and uniformly from its correct and incorrect responses, respectively. We measure within-question discrimination between correct and incorrect responses to the same question, and cross-question discrimination between questions with different empirical accuracies for the same model:

$$\text{within-question AUROC} = \Pr(S_{qi^+} > S_{qi^-}) + \tfrac{1}{2} \Pr(S_{qi^+} = S_{qi^-}), \tag{4}$$

$$\text{cross-question C-Index} = \Pr(\bar{S}_q > \bar{S}_{q'} \mid \hat{r}_q > \hat{r}_{q'}) + \tfrac{1}{2} \Pr(\bar{S}_q = \bar{S}_{q'} \mid \hat{r}_q > \hat{r}_{q'}). \tag{5}$$

For within-question AUROC, we average per-question AUROC equally over questions containing both correct and incorrect responses and exclude single-class questions. For cross-question C-Index, we weight comparable pairs equally and exclude pairs tied in empirical accuracy. Both metrics assign half credit to score ties. Empirical accuracy depends on the model and sampled responses.

3.2 Experiment Setup

Datasets and models. We evaluate five models—Qwen3-8B/32B (Yang et al., 2025a), GPT-OSS-20B (OpenAI, 2025), Gemma 4-E4B, and Gemma 4-26B-A4B (Gemma Team, 2026)—on mathematical reasoning with AIME 2024/2025 (Balunović et al., 2025) and scientific question answering with GPQA-Diamond (Rein et al., 2023). We refer to Gemma 4-26B-A4B as Gemma4-26B throughout the rest of this paper. For each model and question, we apply different scoring choices to the same generated responses and evaluate them against the same correctness labels. We report results across models and datasets to distinguish shared patterns from effects that depend on a particular model or task. Detailed comparisons and settings accompany the corresponding analyses.

We use these evaluation metrics to compare four certainty scores. SC, NegEnt, and MaxLP from Section 2 summarize concentration in the next-token distribution. To examine how certainty varies with candidate rank, we introduce the logit gap. Let z_k denote the logit corresponding to probability p_k, where tokens are ordered by decreasing probability and indexed from zero. The gap between the most probable token and the token indexed by k is

$$\Delta_k = \log p_0 - \log p_k = z_0 - z_k, \quad 1 \le k \le K - 1. \tag{6}$$

Varying k allows us to compare the discriminative power of gaps at different ranks. Our main four-metric comparison uses the top 20 token probabilities, with logit gap measured by Δ_{19}, the difference between the logits of the most probable and twentieth-most probable tokens. We compare the four scores on both within-question and cross-question targets, examine how local generation patterns influence these scores, and study how scoring different semantic segments and reasoning positions changes discrimination.

4 What Does Token-Level Certainty Predict?

In this section, we ask what token-level certainty tells us about a generated response and its question. To interpret this information, we examine how certainty relates to response correctness and question-level empirical accuracy, and how scores vary with local token characteristics. This helps clarify what a high certainty score indicates and how its interpretation depends on the measurement. Unless otherwise specified, we score each response by averaging its token scores.

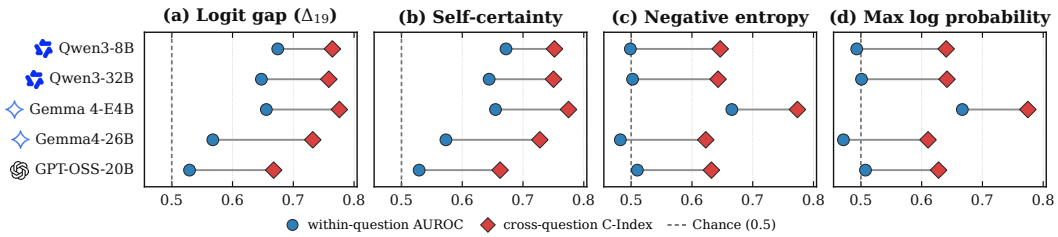


Figure 2: Within-question and cross-question discrimination across five models and four certainty metrics. Each panel shows one metric, with rows linking each model's within-question AUROC (blue circles) and cross-question C-Index (red diamonds). Points average results equally over AIME 2024, AIME 2025, and GPQA-Diamond.

4.1 Certainty's Predictive Value Depends on the Target

Figure 2 compares the four certainty metrics on both prediction targets, with results averaged equally across the three datasets for each model. Cross-question discrimination is higher than within-question discrimination for every evaluated model–metric pair, although the absolute level of discrimination varies substantially across models. GPT-OSS-20B has near-chance within-question discrimination under all four metrics despite retaining a clearer cross-question signal. Gemma 4-E4B has comparatively strong discrimination on both targets. Discrimination also varies across certainty metrics for the same model and prediction target. For both Qwen models and Gemma4-26B, logit gap and self-certainty yield higher discrimination on both targets than negative entropy and maximum log probability; the latter two give within-question AUROC values near or below chance. Gemma 4-E4B shows little variation across metrics. Appendix A provides paired confidence intervals, compares the two targets on a common question set, and examines sensitivity to candidate rank.

These results show that the predictive value of token-level certainty depends on the target. In our comparisons, it distinguishes differences in empirical accuracy across questions more strongly than correctness differences among responses to the same question, with within-question discrimination varying substantially across models and metrics. This distinction bears on the conditions under which certainty can support response selection (Kang et al., 2025), likelihood-based action selection (Wang et al., 2024a), and reinforcement learning with relative self-certainty among responses to the same question (Zhao et al., 2025). These uses rely on scores to identify outputs worth selecting or reinforcing for a given input. The differences we observe across models suggest that the reliability of these scores may depend on the model. Meanwhile, the stronger cross-question signal provides information about which questions a model is more likely to answer correctly. Work on allocating computation according to question difficulty (Wang et al., 2024b) and routing questions between models (Chuang et al., 2025) demonstrates the value of decisions at the question level. Our findings motivate further study of how certainty obtained directly during generation can inform such decisions.

4.2 Certainty Reflects Local Token Characteristics

The preceding comparisons show that certainty's ranking ability depends on the prediction target, model, and metric. To examine what else certainty reflects, we now consider its association with local token characteristics. Specifically, we ask whether token categories and positions within words exhibit systematic differences in certainty that contribute to the scores aggregated over a response.

Token Categories. We compare four categories defined by the characters in each token: Semantic tokens contain word-like text; Structural tokens represent formatting or punctuation; Symbolic tokens contain mathematical symbols or variables without digits; and Digit tokens contain digits. Figure 3 shows that category shares differ and vary across models. Structural, Symbolic, and Digit tokens have higher mean scores than Semantic tokens across all evaluated models and metrics. Differences in category proportions can therefore change responses' average certainty scores. Appendix B.1 provides classification details and further analyses.

Positions within Words. A word can be split into tokens called subwords (Clark et al., 2025). Figure 4 compares certainty at later subword positions with the first position, grouping words by token count.

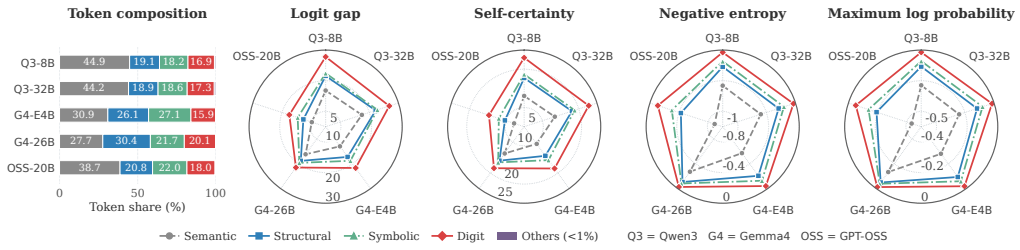


Figure 3: Structural, Symbolic, and Digit tokens have higher mean certainty than Semantic tokens across four metrics and five models. The left panel shows each category's share of all tokens, including Others (below 1% for each model); the four radar panels show category means. Radar axes denote models; colors denote token categories. We average shares and means equally over the three datasets. The model abbreviations G4-E4B and G4-26B denote Gemma 4-E4B and Gemma4-26B, respectively, in both the category-share panel and the radar panels.

Later subwords have higher mean scores than the first across metrics and models, although the differences need not increase at every position. Higher certainty may reflect word completion: the first subword helps select a word, while later pieces complete its spelling or grammatical form given the preceding pieces. Removing later pieces of detected multi-token words and numbers, while leaving other tokens unchanged, does not consistently improve ranking based on the average logit gap of the retained tokens; changes vary with the prediction target, model, and dataset (Appendix B.2).

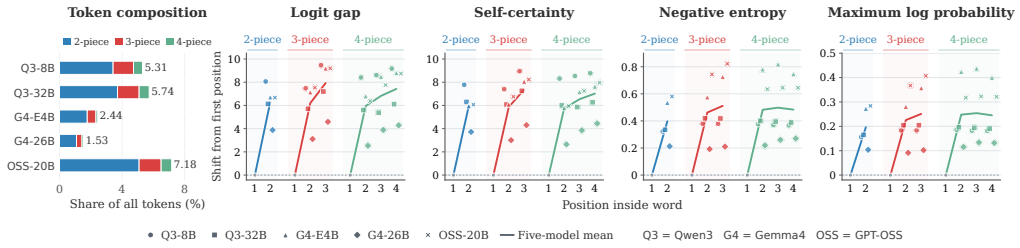


Figure 4: Later subwords have higher mean certainty than the first across the four metrics and five models shown. Colors distinguish words split into 2, 3, or 4 tokens. The left panel shows these detected words' share of all token occurrences. Model symbols in the four right panels show each position's mean minus the first-position mean for the same word length. Positive differences show higher certainty after the first subword across metrics and word lengths. The model abbreviations G4-E4B and G4-26B denote Gemma 4-E4B and Gemma4-26B, respectively.

5 Where Do Effective Signals Concentrate?

Section 4 examined the signals in token-level certainty. These signals contain information both relevant and irrelevant to our prediction targets. We therefore examine how aggregating them across response regions can improve prediction. Prior work has aggregated certainty within response segments, as in the trial-answer scoring used by DEER (Yang et al., 2025b) and CGR (Nogueira et al., 2025). Other work has aggregated certainty at different positions in reasoning traces, including tail tokens and local windows in DeepConf (Fu et al., 2025) and prefixes in Kim & Kim (2026). However, selection accuracy and compute savings also depend on candidate responses and decision rules, so they alone cannot establish local certainty's discriminative ability for different prediction targets (Section 3.1). We therefore hold the certainty metric, generated traces, and correctness labels fixed while changing only the scored region, comparing reasoning, summary, and final-answer segments as well as matched-length prefix and suffix windows. By measuring discrimination for within-question correctness and cross-question empirical accuracy across models and tasks, we examine which

regions better distinguish candidate responses to the same question, which better indicate question difficulty, and whether these advantages are consistent.

5.1 Reasoning Segments

We compare three regions of each trace: reasoning, summary, and final-answer tokens. The model's output-format boundaries define the reasoning and summary regions. In Qwen3 (Yang et al., 2025a), for example, reasoning lies between <think> and </think>, and summary denotes the response after the closing marker. Final-answer tokens are the extracted answer span within that response, so the answer and summary regions overlap. We average logit gap over each region and evaluate the resulting scores using within-question AUROC and cross-question C-Index (Figure 5). This comparison tests whether retaining the reasoning, the subsequent response, or only the answer changes certainty's ability to discriminate between questions or responses.

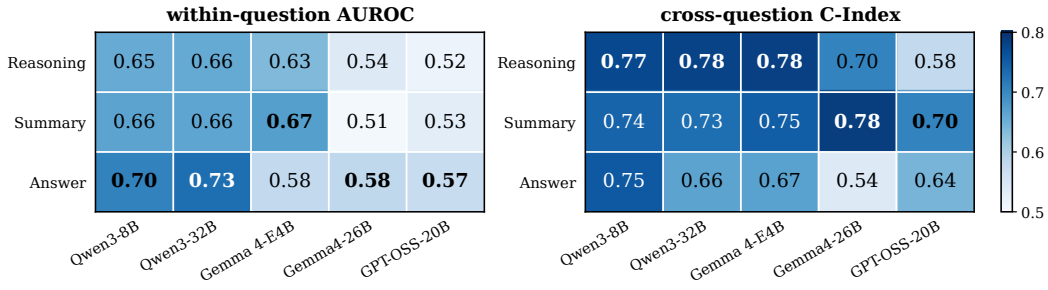


Figure 5: Segment selection changes the target captured by certainty. Cells show each model's logit-gap ranking scores for reasoning, summary, and answer tokens, averaged equally across the three datasets. Panels use the same color scale; boldface indicates the highest score for each model and prediction target. Answer tokens more often favor within-question ranking, whereas reasoning or summary tokens more often favor cross-question ranking.

Segment choice affects the two targets differently. With scores averaged across datasets within each model, answer tokens give the highest within-question AUROC for both Qwen models, Gemma4-26B, and GPT-OSS-20B; Gemma 4-E4B instead favors the summary. The overall answer-token AUROC is 0.633, indicating that even the strongest average segment score provides limited correctness discrimination. For cross-question ranking, every model favors reasoning or summary over answer-only scoring. Restricting the score to the final answer therefore often helps distinguish alternative responses to the same question, while losing information useful for ranking questions by empirical accuracy. The preferred segment therefore depends on the prediction target. Because the segments differ in length and position, we compare temporal windows to locate each signal along a trace.

5.2 Temporal Localization

We next compare prefix and suffix windows of matched token lengths to examine where certainty is informative without relying on model-specific segment boundaries. For each window length, we average the logit gap over the selected tokens and evaluate within-question AUROC and cross-question C-Index (Figure 6). Traces shorter than the specified window are scored in full. Appendix C examines whether the same trends hold for windows defined as a fraction of trace length.

For cross-question ranking, the average C-Index rises rapidly as either prefix or suffix windows expand and largely plateaus at around 1,000 tokens. Both prefixes and suffixes therefore contain information for ranking questions by empirical accuracy. In particular, prefix scores provide an informative ranking early in generation. Thus, even before a response is complete, its opening tokens can provide an initial indication of which questions the model is more likely to answer correctly.

For within-question trace selection, the temporal asymmetry is sharper. Short prefixes have little discriminative power, with AUROC close to chance; the signal strengthens as the observed window moves toward the end of the trace. Suffix windows, especially short suffixes, are already more informative than prefix windows of the same length, and some suffix aggregations exceed the full-

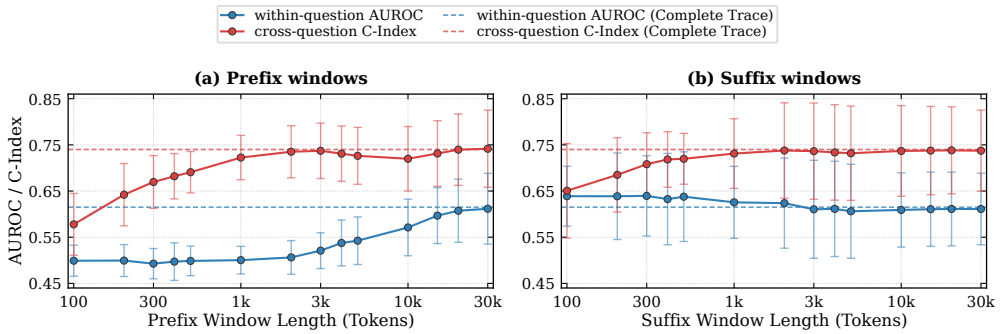


Figure 6: Temporal windows locate certainty signals within a trace. Panels (a) and (b) show absolute prefix and suffix windows, respectively. Averaging logit gap over early prefixes quickly recovers cross-question accuracy ranking, while early within-question AUROC remains near chance. Short suffix windows already provide a stronger within-question trace-correctness signal. Lines show averages across five models and three datasets; error bars show the standard deviation of the scores across the evaluated combinations of models and datasets.

trace average. This suggests that trace-level correctness evidence is more concentrated near the final answer, where the model commits to and resolves a specific sampled solution. Comparisons across all four certainty metrics likewise show informative prefixes for cross-question ranking and higher average within-question AUROC for suffixes than for full traces (Appendix C.3). Together, these findings motivate the division of operations in TACS (Section 6). Earlier or wider-window certainty is better matched to cross-question budget allocation, while late-window certainty is better suited to post-generation candidate filtering or weighting.

6 How Can Certainty Signals Guide Test-Time Compute?

Our analyses show that token-level certainty's discriminative power varies with the prediction target, scoring metric, and aggregation region. We translate these findings into Target-Aligned Certainty Scaling (TACS), a test-time scaling (TTS) framework that selects a scoring metric and matches signals from different regions to the compute decisions they support. We use logit gap because it performs well in the preceding ranking comparisons, is inexpensive to compute directly from logit differences, and preserves rankings under uniform temperature scaling. Guided by the differences in strength and localization between cross-question and within-question signals, TACS uses early certainty to allocate sampling budgets and late certainty to weight candidate traces' votes. We first test the contributions of early allocation and late vote weighting independently, then evaluate the combined framework's trade-off between generated-token cost and answer accuracy.

Early allocation. Our analyses show that certainty better distinguishes empirical performance across questions and that short prefixes already capture a strong cross-question signal (Section 5). We can therefore control each question's sample count from prefix scores alone. A trace counts toward the sampling target if its mean logit gap over the first 1,000 tokens is at least θ_m, a model-specific threshold that is calibrated on a separate dataset and held fixed across the evaluation datasets. Appendix D.1 reports the model-specific thresholds. We fix $N = 32$ throughout this section, sampling until 32 traces meet the threshold or we observe 256 traces. This shared target yields different sample counts across questions: those that more often produce high-certainty prefixes reach the target with fewer samples, whereas others continue sampling. For all five models, the hardest question group receives more samples on average than the easiest group (Appendix D.3). We compare prefix allocation with fixed 256-trace sampling across voting methods, using the same candidate set for all methods under each strategy. In Table 1, prefix allocation reduces generated-token cost by 82.44%, while overall accuracy decreases by only 0.17–0.33 percentage points (pp) relative to fixed-budget sampling when both sampling strategies use the same voting method.

Late voting. Within-question information is more concentrated near the end of a trace (Section 5), motivating suffix certainty as an auxiliary ranking signal for voting. We rank sampled responses with

Table 1: Answer aggregation under fixed-budget sampling and early allocation. We compare majority voting with rank-weighted voting using certainty computed over the full trace or the final 1,000 or 500 tokens (Suf. 1k and Suf. 500, respectively). Within each sampling run, all voting methods share candidate responses and thus generated-token cost. Rank-weighted voting based on the final 500 tokens has the highest overall accuracy among the four methods under both sampling conditions. With early allocation, it reaches 79.54% accuracy at $0.176\times$ the token cost of fixed 256-trace sampling.

	Fixed Budget					**Early allocation**				
	Accuracy (%) ↑				Cost ↓	Accuracy (%) ↑				Cost ↓
Model	Majority	Full	Suf. 1k	Suf. 500		Majority	Full	Suf. 1k	Suf. 500	
Qwen3-8B	75.43	76.15	76.36	**76.45**	1.000	74.96	75.73	76.21	**76.33**	0.142
Qwen3-32B	79.63	81.11	80.78	**81.31**	1.000	80.68	81.38	81.37	**81.75**	0.145
Gemma4-E4B	62.11	63.72	63.64	**64.38**	1.000	61.22	62.92	63.10	**63.80**	0.186
Gemma4-26B	91.39	**91.51**	91.37	91.49	1.000	90.87	**91.13**	90.71	90.86	0.192
GPT-OSS-20B	**84.98**	84.78	84.97	84.94	1.000	84.46	84.48	84.87	**84.96**	0.214
Overall	78.71	79.45	79.42	**79.71**	1.000	78.44	79.13	79.25	**79.54**	0.176

valid answers by their mean logit gap over the final 500 tokens. Each response receives a vote weight equal to its ascending certainty rank, so higher-certainty responses receive larger weights. Responses below the prefix threshold also participate in voting, and the answer with the largest total vote weight wins. This leaves sampling cost unchanged. With prefix allocation and fixed 256-trace sampling, rank-weighted voting based on certainty over the final 500 tokens improves overall accuracy over majority voting by 1.10 and 1.01 pp, respectively. Four of five models improve under each condition (Table 1). Compared with full-trace rank-weighted voting, the gains are 0.41 and 0.26 pp, respectively, although Gemma4-26B shows the reverse ordering under both conditions. These results echo our earlier observations: the relatively weak within-question signal is stronger near the end of a trace, while gains in answer aggregation remain model-dependent.

Main results. We evaluate TACS against a fixed-budget self-consistency baseline, cons@256, which uses majority voting over 256 traces. As shown in Table 1, TACS reaches 79.54% accuracy at $0.176\times$ the token cost of cons@256. Its accuracy exceeds fixed-256 majority voting by 0.83 pp and is 0.17 pp below fixed-256 sampling with the same voting rule based on certainty over the final 500 tokens. Appendix D.4 provides additional comparisons with fixed-budget and adaptive baselines. TACS thus uses early cross-question information for sampling decisions and late within-question information for vote weighting, aligning scoring regions with prediction targets.

7 Conclusion

Our experiments show that token-level certainty generally distinguishes questions by empirical accuracy more effectively than it distinguishes correct from incorrect responses to the same question. Its predictive value varies across models, metrics, and response regions, while its scores also reflect local token characteristics. Question-level ranking information emerges early, whereas within-question correctness information is more concentrated near the end. TACS uses this distinction to allocate samples with prefix certainty and weight answer votes with suffix certainty. In our experiments, TACS reduces generated-token cost by 82.4% while increasing overall accuracy from 78.71% to 79.54% relative to majority voting over 256 responses. These findings support matching the scoring region to the prediction target and evaluating certainty both as a predictor and as part of the decision rule that uses it. We believe these insights can guide more effective uses of certainty across LLM training and inference.

AI USE STATEMENT

We used generative AI tools only to assist with writing code and with language polishing and proofreading of the final manuscript. We did not use generative AI tools to formulate the research questions or hypotheses, develop the conceptual or theoretical framework, design the methodology or experiments, generate or clean datasets, derive mathematical claims or proofs, analyze data, or interpret results. All AI-assisted code was reviewed and tested by the authors, and every AI-assisted textual edit was checked against the authors' intended meaning. We take full responsibility for the final content of this work, including all text, claims, code, and other artifacts.

ETHICS STATEMENT

This work uses existing public reasoning benchmarks and model-generated traces and does not involve human subjects, personally identifiable information, or deployment in high-stakes settings. We did not identify ethical concerns specific to this study beyond the general computational resource use and potential downstream misuse associated with large language model research.

REPRODUCIBILITY STATEMENT

Additional diagnostic and policy analyses are provided in Appendices A–D. Appendix E documents the exact prompt templates, model-specific chat settings, decoding parameters, sample counts, random seed, trace collection procedures, and the TACS evaluation protocol. We plan to release the implementation code and scripts used for data processing, experiments, and evaluation publicly after review.

REFERENCES

Shivam Agarwal, Zimin Zhang, Lifan Yuan, Jiawei Han, and Hao Peng. The Unreasonable Effectiveness of Entropy Minimization in LLM Reasoning, 2025. URL https://arxiv.org/abs/2505.15134.

Pranjal Aggarwal, Aman Madaan, Yiming Yang, and Mausam. Let's sample step by step: Adaptive-consistency for efficient reasoning and coding with llms. In *Conference on Empirical Methods in Natural Language Processing*, 2023. URL https://www.semanticscholar.org/paper/b4a6c010724f0459c9791018e34a982cf96987cf.

Mislav Balunović, Jasper Dekoninck, Ivo Petrov, Nikola Jovanović, and Martin Vechev. Matharena: Evaluating llms on uncontaminated math competitions. In *arXiv.org*, 2025. URL https://arxiv.org/abs/2505.23281.

Yu-Neng Chuang, Prathusha Kameswara Sarma, Parikshit Gopalan, John Boccio, Sara Bolouki, Xia Hu, and Helen Zhou. Learning to route llms with confidence tokens. In *International Conference on Machine Learning*, 2025. URL https://proceedings.mlr.press/v267/chuang25b.html.

Christian Clark, Byung-Doh Oh, and William Schuler. How well does first-token entropy approximate word entropy as a psycholinguistic predictor?, 2025. URL https://arxiv.org/abs/2507.22209.

Yichao Fu, Xuewei Wang, Yuandong Tian, and Jiawei Zhao. Deep think with confidence. In *arXiv.org*, 2025. URL https://arxiv.org/abs/2508.15260.

Gemma Team. Gemma 4 Technical Report, 2026. URL https://arxiv.org/abs/2607.02770.

Zhewei Kang, Xuandong Zhao, and Dawn Song. Scalable best-of-n selection for large language models via self-certainty. In *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 19720–19745. Curran Associates, Inc., 2025. doi: 10.52202/085713-0663. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/1c7eff166a8e345f664f0faa8f4e4d2e-Paper-Conference.pdf.

Hojin Kim and Jaehyung Kim. Reasoning or fluency? dissecting probabilistic confidence in best-of-n selection. In *arXiv.org*, 2026. URL https://arxiv.org/abs/2601.13735.

Junseok Kim, Nakyeong Yang, Kyungmin Min, and Kyomin Jung. Reliability-Aware Adaptive Self-Consistency for Efficient Sampling in LLM Reasoning, 2026. URL https://arxiv.org/abs/2601.02970.

Yiwei Li, Peiwen Yuan, Shaoxiong Feng, Boyuan Pan, Xinglin Wang, Bin Sun, Heda Wang, and Kan Li. Escape sky-high cost: Early-stopping self-consistency for multi-step reasoning. In *International Conference on Learning Representations*, 2024. URL https://www.semanticscholar.org/paper/36260206be303b3820df8fbba161591da0ca987c.

João Paulo Nogueira, Wentao Sun, Alonso Silva, and Laith Zumot. Certainty-guided reasoning in large language models: A dynamic thinking budget approach. In *arXiv.org*, 2025. URL https://arxiv.org/abs/2509.07820.

OpenAI. gpt-oss-120b & gpt-oss-20b Model Card, 2025. URL https://arxiv.org/abs/2508.10925.

Mihir Prabhudesai, Lili Chen, Alex Ippoliti, Katerina Fragkiadaki, Hao Liu, and Deepak Pathak. Maximizing Confidence Alone Improves Reasoning, 2025. URL https://arxiv.org/abs/2505.22660.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. Gpqa: A graduate-level google-proof q&a benchmark. In *arXiv.org*, 2023. URL https://www.semanticscholar.org/paper/210b0a3d76e93079cc51b03c4115fde545eea966.

Aman Sharma and Paras Chopra. Think Just Enough: Sequence-Level Entropy as a Confidence Signal for LLM Reasoning, 2025. URL https://arxiv.org/abs/2510.08146.

Amir Taubenfeld, Tom Sheffer, Eran Ofek, Amir Feder, Ariel Goldstein, Zorik Gekhman, and Gal Yona. Confidence improves self-consistency in llms. In *arXiv.org*, 2025. URL https://arxiv.org/abs/2502.06233.

Han Wang, Archiki Prasad, Elias Stengel-Eskin, and Mohit Bansal. Soft self-consistency improves language model agents. In *arXiv.org*, 2024a. URL https://www.semanticscholar.org/paper/04b052f14d683618b87e1d11b4401f16e4055303.

Shenzhi Wang, Le Yu, Chang Gao, Chujie Zheng, Shixuan Liu, Rui Lu, Kai Dang, Xionghui Chen, Jianxin Yang, Zhenru Zhang, Yuqiong Liu, An Yang, Andrew Zhao, Yang Yue, Shiji Song, Bowen Yu, Gao Huang, and Junyang Lin. Beyond the 80/20 rule: High-entropy minority tokens drive effective reinforcement learning for llm reasoning, 2025. URL https://arxiv.org/abs/2506.01939.

Xinglin Wang, Shaoxiong Feng, Yiwei Li, Peiwen Yuan, Yueqi Zhang, Chuyi Tan, Boyuan Pan, Yao Hu, and Kan Li. Make every penny count: Difficulty-adaptive self-consistency for cost-efficient reasoning. In *arXiv.org*, 2024b. URL https://arxiv.org/abs/2408.13457.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=1PL1NIMMrw.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025a. URL https://arxiv.org/abs/2505.09388. arXiv:2505.09388.

Chenxu Yang, Qingyi Si, Yongjie Duan, Zheliang Zhu, Chenyu Zhu, Zheng Lin, Li Cao, and Weiping Wang. Dynamic early exit in reasoning models. In *arXiv.org*, 2025b. URL https://arxiv.org/abs/2504.15895v1.

Xuandong Zhao, Zhewei Kang, Aosong Feng, Sergey Levine, and D. Song. Learning to reason without external rewards. In *arXiv.org*, 2025. URL https://www.semanticscholar.org/paper/89e3017e1ffdf5f7cedd3b5bb08fcc0ab48200ba.

A Details for Section 4.1

The analyses in Appendices A.1–A.3 use the same full-trace scores as Figure 2: logit gap Δ_{19}, self-certainty, negative entropy, and maximum log probability, computed from the full-softmax probabilities of the 20 most probable tokens, without renormalization. For each complete response, we average its token scores. All four metrics use the same set of responses; Appendix E.2 reports the sample sizes for these response-level comparisons.

A.1 Discrimination across Models and Datasets

Tables 2 and 3 report within-question AUROC and cross-question C-Index as defined in Section 3.1. We compute statistics within each model–dataset setting and report equal-weight averages by model, by dataset, and overall. Cross-question C-Index exceeds within-question AUROC in all 60 model–dataset–metric combinations. Paired 95% intervals for the difference are above zero in 34 of the 60 combinations and in all model-level, dataset-level, and overall averages. Logit gap and self-certainty have higher within-question discrimination on the two AIME datasets than on GPQA-Diamond, whereas negative entropy and maximum log probability show the opposite ordering. The cross-question and within-question statistics concern distinct prediction targets.

A.2 Correctness Comparisons within and across Questions

To hold the prediction outcome fixed, we compare correctness concordance for same-question and different-question trace pairs, denoted by A_{same} and A_{diff}. Pairs are equally weighted within each group, with half credit for score ties. If the corresponding correct–incorrect pair counts are N_{same} and N_{diff}, pooled AUROC within a model–dataset setting is

$$\text{AUROC}_{\text{pooled}} = \frac{N_{\text{same}} A_{\text{same}} + N_{\text{diff}} A_{\text{diff}}}{N_{\text{same}} + N_{\text{diff}}}. \tag{7}$$

Unlike the question-equal within-question AUROC, A_{same} weights each question by its number of correct–incorrect pairs. A_{diff} compares individual trace scores against correctness labels; unlike cross-question C-Index, it does not compare mean question scores against empirical accuracy.

Tables 4 and 5 report concordance for both groups and pooled AUROC. Different-question concordance exceeds same-question concordance in all 60 model–dataset–metric combinations. Paired 95% intervals for this difference are above zero in 47 combinations. For logit gap, the overall values of same-question concordance, different-question concordance, and pooled AUROC are 0.583, 0.801, and 0.799, respectively. Different-question pairs account for 99.05% of correct–incorrect pairs on average across settings, so pooled AUROC can mask weak within-question discrimination.

A.3 Comparing Both Targets on the Same Question Set

To compare both targets on the same question set, we retain only questions with both correct and incorrect responses for each model, yielding 865 of 1,290 model–question groups. Within-question AUROC is unchanged; we recompute cross-question C-Index on this restricted set using the same rules for evaluating discrimination. Tables 6 and 7 report the results.

Restricting both targets to the same questions lowers the overall cross-question C-Index for all four metrics; for logit gap, it falls from 0.740 to 0.666. On AIME 2024 and GPQA-Diamond, Gemma4-26B has lower cross-question C-Index than within-question AUROC for all four metrics; difference intervals cross zero. Overall difference intervals include zero for logit gap and self-certainty but remain above zero for the other two metrics. Thus, question selection partly accounts for the cross-question advantage and changes both the range of empirical accuracies among the retained questions and the number of questions available for comparison.

Table 2: Full-trace within-question AUROC and cross-question C-Index by model. Cross-question discrimination is higher for every model–metric average. Model rows average three datasets equally; Overall averages all 15 settings equally. Evaluation follows Section 3.1. Gemma4-4B denotes Gemma 4-E4B. Blue "Yes" denotes a positive difference, not statistical significance. Brackets give pointwise 95% paired question-bootstrap confidence intervals (Appendix E.2).

Model	within-question AUROC (W)	cross-question C-Index (C)	C − W 95% CI	cross-question > within-question?
A. Logit gap (Δ_{19})				
Qwen3-8B	0.674	0.764	+0.0899 [0.0396, 0.1424]	Yes
Qwen3-32B	0.647	0.758	+0.1109 [0.0549, 0.1631]	Yes
Gemma4-4B	0.656	0.776	+0.1202 [0.0565, 0.1860]	Yes
Gemma4-26B	0.568	0.732	+0.1645 [0.0603, 0.2741]	Yes
GPT-OSS-20B	0.529	0.668	+0.1385 [0.0686, 0.2041]	Yes
Overall	0.615	0.740	+0.1248 [0.0829, 0.1677]	Yes
B. Self-certainty				
Qwen3-8B	0.672	0.752	+0.0798 [0.0244, 0.1354]	Yes
Qwen3-32B	0.644	0.750	+0.1059 [0.0492, 0.1589]	Yes
Gemma4-4B	0.655	0.775	+0.1202 [0.0553, 0.1867]	Yes
Gemma4-26B	0.573	0.728	+0.1545 [0.0523, 0.2660]	Yes
GPT-OSS-20B	0.529	0.662	+0.1332 [0.0634, 0.1993]	Yes
Overall	0.615	0.733	+0.1187 [0.0755, 0.1613]	Yes
C. Negative entropy				
Qwen3-8B	0.499	0.647	+0.1479 [0.0693, 0.2248]	Yes
Qwen3-32B	0.502	0.643	+0.1407 [0.0726, 0.2036]	Yes
Gemma4-4B	0.666	0.774	+0.1080 [0.0407, 0.1814]	Yes
Gemma4-26B	0.482	0.623	+0.1406 [0.0349, 0.2476]	Yes
GPT-OSS-20B	0.510	0.632	+0.1218 [0.0472, 0.1919]	Yes
Overall	0.532	0.664	+0.1318 [0.0888, 0.1758]	Yes
D. Maximum log probability				
Qwen3-8B	0.493	0.640	+0.1472 [0.0669, 0.2234]	Yes
Qwen3-32B	0.501	0.642	+0.1407 [0.0733, 0.2039]	Yes
Gemma4-4B	0.667	0.775	+0.1079 [0.0421, 0.1800]	Yes
Gemma4-26B	0.471	0.610	+0.1393 [0.0355, 0.2465]	Yes
GPT-OSS-20B	0.508	0.628	+0.1203 [0.0458, 0.1897]	Yes
Overall	0.528	0.659	+0.1311 [0.0886, 0.1744]	Yes

Table 3: Full-trace within-question AUROC and cross-question C-Index by dataset. Cross-question discrimination is higher for every dataset–metric average. Dataset rows average five models equally; Overall averages all 15 settings equally. Evaluation follows Section 3.1. Blue "Yes" denotes a positive difference, not statistical significance. Brackets give pointwise 95% intervals obtained by paired bootstrap resampling of questions (Appendix E.2).

Dataset	**within-question** AUROC (W)	**cross-question** C-Index (C)	**C − W** 95% CI	**cross-question > within-question?**
A. Logit gap (Δ_{19})				
AIME 2024	0.624	0.759	+0.1348 [0.0548, 0.2138]	Yes
AIME 2025	0.637	0.783	+0.1458 [0.0500, 0.2373]	Yes
GPQA-Diamond	0.583	0.677	+0.0938 [0.0550, 0.1309]	Yes
Overall	0.615	0.740	+0.1248 [0.0829, 0.1677]	Yes
B. Self-certainty				
AIME 2024	0.623	0.749	+0.1265 [0.0464, 0.2033]	Yes
AIME 2025	0.638	0.778	+0.1396 [0.0429, 0.2322]	Yes
GPQA-Diamond	0.583	0.673	+0.0900 [0.0508, 0.1282]	Yes
Overall	0.615	0.733	+0.1187 [0.0755, 0.1613]	Yes
C. Negative entropy				
AIME 2024	0.522	0.692	+0.1701 [0.0922, 0.2489]	Yes
AIME 2025	0.526	0.657	+0.1317 [0.0402, 0.2264]	Yes
GPQA-Diamond	0.547	0.641	+0.0936 [0.0465, 0.1365]	Yes
Overall	0.532	0.664	+0.1318 [0.0888, 0.1758]	Yes
D. Maximum log probability				
AIME 2024	0.518	0.685	+0.1671 [0.0887, 0.2495]	Yes
AIME 2025	0.520	0.652	+0.1326 [0.0419, 0.2283]	Yes
GPQA-Diamond	0.546	0.639	+0.0935 [0.0466, 0.1361]	Yes
Overall	0.528	0.659	+0.1311 [0.0886, 0.1744]	Yes

Table 4: Correctness discrimination by model under equal trace-pair weighting. Different-question concordance exceeds same-question concordance for every model–metric average. Model rows average three datasets equally; Overall averages all 15 settings equally. A_{same} and A_{diff} measure correctness concordance for equally weighted trace pairs from the same and different questions, respectively. Pooled AUROC is computed within each setting using Equation 7 before averaging. Blue "Yes" denotes a positive difference, not statistical significance. Brackets give pointwise 95% paired question-bootstrap intervals (Appendix E.2).

Model	Same-question concordance (A_{same})	Different-question concordance (A_{diff})	Pooled AUROC	Difference $A_{\text{diff}} - A_{\text{same}}$ [95% CI]	Different > Same? Point estimate
A. Logit gap (Δ_{19})					
Qwen3-8B	0.633	0.869	0.866	+0.2360 [0.1674, 0.3033]	Yes
Qwen3-32B	0.642	0.848	0.846	+0.2061 [0.1406, 0.2757]	Yes
Gemma4-4B	0.619	0.839	0.836	+0.2195 [0.1319, 0.3088]	Yes
Gemma4-26B	0.498	0.766	0.763	+0.2682 [0.1169, 0.3454]	Yes
GPT-OSS-20B	0.526	0.686	0.684	+0.1601 [0.0887, 0.2390]	Yes
Overall	0.583	0.801	0.799	+0.2180 [0.1584, 0.2638]	Yes
B. Self-certainty					
Qwen3-8B	0.632	0.865	0.862	+0.2326 [0.1638, 0.3015]	Yes
Qwen3-32B	0.642	0.846	0.844	+0.2047 [0.1398, 0.2743]	Yes
Gemma4-4B	0.618	0.838	0.835	+0.2195 [0.1314, 0.3084]	Yes
Gemma4-26B	0.501	0.763	0.760	+0.2627 [0.1102, 0.3399]	Yes
GPT-OSS-20B	0.526	0.682	0.680	+0.1563 [0.0836, 0.2347]	Yes
Overall	0.584	0.799	0.797	+0.2152 [0.1554, 0.2614]	Yes
C. Negative entropy					
Qwen3-8B	0.482	0.739	0.736	+0.2567 [0.1802, 0.3270]	Yes
Qwen3-32B	0.509	0.739	0.737	+0.2303 [0.1683, 0.3004]	Yes
Gemma4-4B	0.622	0.836	0.833	+0.2140 [0.1257, 0.3055]	Yes
Gemma4-26B	0.422	0.590	0.588	+0.1673 [0.0414, 0.2428]	Yes
GPT-OSS-20B	0.513	0.640	0.639	+0.1279 [0.0520, 0.2081]	Yes
Overall	0.509	0.709	0.707	+0.1992 [0.1479, 0.2433]	Yes
D. Maximum log probability					
Qwen3-8B	0.479	0.730	0.728	+0.2514 [0.1755, 0.3220]	Yes
Qwen3-32B	0.505	0.733	0.731	+0.2276 [0.1648, 0.2974]	Yes
Gemma4-4B	0.624	0.837	0.835	+0.2124 [0.1245, 0.3037]	Yes
Gemma4-26B	0.416	0.564	0.562	+0.1483 [0.0275, 0.2241]	Yes
GPT-OSS-20B	0.511	0.633	0.631	+0.1223 [0.0454, 0.2025]	Yes
Overall	0.507	0.699	0.697	+0.1924 [0.1422, 0.2367]	Yes

Table 5: Correctness discrimination by dataset under equal trace-pair weighting. Different-question concordance exceeds same-question concordance for every dataset–metric average. Dataset rows average five models equally; Overall averages all 15 settings equally. A_{same} and A_{diff} measure correctness concordance for equally weighted trace pairs from the same and different questions, respectively. Pooled AUROC is computed within each setting using Equation 7 before averaging. Blue "Yes" denotes a positive difference, not statistical significance. Brackets give pointwise 95% paired question-bootstrap intervals (Appendix E.2).

	Same-question	**Different-question**			
Dataset	concordance (A_{same})	concordance (A_{diff})	**Pooled** AUROC	**Difference** $A_{\text{diff}} - A_{\text{same}}$ [95% CI]	**Different > Same?** Point estimate
A. Logit gap (Δ_{19})					
AIME 2024	0.598	0.854	0.852	+0.2563 [0.1435, 0.3337]	Yes
AIME 2025	0.587	0.837	0.833	+0.2508 [0.1208, 0.3512]	Yes
GPQA-Diamond	0.565	0.712	0.712	+0.1468 [0.0925, 0.1956]	Yes
Overall	0.583	0.801	0.799	+0.2180 [0.1584, 0.2638]	Yes
B. Self-certainty					
AIME 2024	0.597	0.851	0.848	+0.2540 [0.1386, 0.3319]	Yes
AIME 2025	0.589	0.837	0.834	+0.2488 [0.1182, 0.3479]	Yes
GPQA-Diamond	0.565	0.708	0.708	+0.1427 [0.0888, 0.1919]	Yes
Overall	0.584	0.799	0.797	+0.2152 [0.1554, 0.2614]	Yes
C. Negative entropy					
AIME 2024	0.491	0.751	0.749	+0.2605 [0.1661, 0.3337]	Yes
AIME 2025	0.504	0.707	0.704	+0.2031 [0.0951, 0.3051]	Yes
GPQA-Diamond	0.534	0.668	0.668	+0.1342 [0.0799, 0.1836]	Yes
Overall	0.509	0.709	0.707	+0.1992 [0.1479, 0.2433]	Yes
D. Maximum log probability					
AIME 2024	0.488	0.739	0.737	+0.2514 [0.1561, 0.3268]	Yes
AIME 2025	0.501	0.694	0.691	+0.1930 [0.0900, 0.2935]	Yes
GPQA-Diamond	0.532	0.665	0.665	+0.1328 [0.0780, 0.1823]	Yes
Overall	0.507	0.699	0.697	+0.1924 [0.1422, 0.2367]	Yes

Table 6: Discrimination on questions with both correct and incorrect responses, by model. Gemma4-26B reverses the cross-question advantage for all four metrics; difference intervals cross zero. Model rows average three datasets equally; Overall averages all 15 settings equally. Both targets use questions with both correct and incorrect responses; within-question AUROC is unchanged. Blue "Yes" denotes a positive difference, not statistical significance. Brackets give pointwise 95% confidence intervals from paired question bootstrapping (Appendix E.2).

Model	**within-question** AUROC (W)	**cross-question** C-Index (C)	**C − W** 95% CI	**cross-question > within-question?**
A. Logit gap (Δ_{19})				
Qwen3-8B	0.674	0.730	+0.0557 [−0.0106, 0.1220]	Yes
Qwen3-32B	0.647	0.734	+0.0866 [0.0279, 0.1422]	Yes
Gemma4-4B	0.656	0.745	+0.0896 [0.0045, 0.1729]	Yes
Gemma4-26B	0.568	0.498	−0.0697 [−0.2793, 0.1676]	No
GPT-OSS-20B	0.529	0.621	+0.0921 [0.0165, 0.1648]	Yes
Overall	0.615	0.666	+0.0509 [−0.0066, 0.1115]	Yes
B. Self-certainty				
Qwen3-8B	0.672	0.726	+0.0539 [−0.0120, 0.1201]	Yes
Qwen3-32B	0.644	0.729	+0.0847 [0.0247, 0.1436]	Yes
Gemma4-4B	0.655	0.740	+0.0857 [−0.0012, 0.1700]	Yes
Gemma4-26B	0.573	0.491	−0.0820 [−0.2773, 0.1474]	No
GPT-OSS-20B	0.529	0.619	+0.0898 [0.0162, 0.1629]	Yes
Overall	0.615	0.661	+0.0464 [−0.0113, 0.1050]	Yes
C. Negative entropy				
Qwen3-8B	0.499	0.656	+0.1577 [0.0750, 0.2402]	Yes
Qwen3-32B	0.502	0.639	+0.1369 [0.0643, 0.2034]	Yes
Gemma4-4B	0.666	0.749	+0.0830 [−0.0020, 0.1649]	Yes
Gemma4-26B	0.482	0.405	−0.0769 [−0.2332, 0.1222]	No
GPT-OSS-20B	0.510	0.590	+0.0793 [−0.0023, 0.1611]	Yes
Overall	0.532	0.608	+0.0760 [0.0215, 0.1340]	Yes
D. Maximum log probability				
Qwen3-8B	0.493	0.650	+0.1571 [0.0739, 0.2391]	Yes
Qwen3-32B	0.501	0.638	+0.1374 [0.0677, 0.2038]	Yes
Gemma4-4B	0.667	0.747	+0.0805 [−0.0057, 0.1630]	Yes
Gemma4-26B	0.471	0.406	−0.0651 [−0.2202, 0.1374]	No
GPT-OSS-20B	0.508	0.586	+0.0785 [−0.0036, 0.1593]	Yes
Overall	0.528	0.606	+0.0777 [0.0223, 0.1352]	Yes

Table 7: Discrimination on questions with both correct and incorrect responses, by dataset. AIME 2024 reverses the cross-question advantage for logit gap and self-certainty; difference intervals cross zero. Dataset rows average five models equally; Overall averages all 15 settings equally. Both targets use questions with both correct and incorrect responses; within-question AUROC is unchanged. Blue "Yes" denotes a positive difference, not statistical significance. Brackets give pointwise 95% confidence intervals from paired question bootstrapping (Appendix E.2).

Dataset	within-question AUROC (W)	cross-question C-Index (C)	C − W 95% CI	cross-question > within-question?
A. Logit gap (Δ_{19})				
AIME 2024	0.624	0.614	−0.0106 [−0.1451, 0.1278]	No
AIME 2025	0.637	0.765	+0.1279 [0.0033, 0.2353]	Yes
GPQA-Diamond	0.583	0.618	+0.0353 [−0.0017, 0.0690]	Yes
Overall	0.615	0.666	+0.0509 [−0.0066, 0.1115]	Yes
B. Self-certainty				
AIME 2024	0.623	0.602	−0.0205 [−0.1466, 0.1147]	No
AIME 2025	0.638	0.764	+0.1263 [−0.0018, 0.2339]	Yes
GPQA-Diamond	0.583	0.616	+0.0335 [−0.0039, 0.0675]	Yes
Overall	0.615	0.661	+0.0464 [−0.0113, 0.1050]	Yes
C. Negative entropy				
AIME 2024	0.522	0.561	+0.0385 [−0.0825, 0.1709]	Yes
AIME 2025	0.526	0.674	+0.1484 [0.0339, 0.2548]	Yes
GPQA-Diamond	0.547	0.589	+0.0412 [0.0017, 0.0795]	Yes
Overall	0.532	0.608	+0.0760 [0.0215, 0.1340]	Yes
D. Maximum log probability				
AIME 2024	0.518	0.560	+0.0414 [−0.0812, 0.1750]	Yes
AIME 2025	0.520	0.669	+0.1493 [0.0357, 0.2553]	Yes
GPQA-Diamond	0.546	0.588	+0.0423 [0.0024, 0.0802]	Yes
Overall	0.528	0.606	+0.0777 [0.0223, 0.1352]	Yes

A.4 Rank Sensitivity

We test how competitor rank affects discrimination using $\Delta_k = z_0 - z_k$ (Section 3.2). Here, k indexes tokens in descending probability order, so Δ_k compares the most probable token with the $(k+1)$th most probable token. This analysis uses a separate response collection with top-100 logits (Appendix E.2). At each evaluated k, we average token-level gaps over each complete response and evaluate both prediction targets on the same responses and labels. We evaluate 21 values of k between 2 and 99, including $k = 19$, which is used in the main comparison.

From $k = 2$ to $k = 99$, within-question AUROC and cross-question C-Index both increase for the Qwen models and change little for the Gemma models. For GPT-OSS-20B, within-question AUROC decreases, whereas cross-question C-Index increases. Across settings, increasing k from 2 to 10 raises mean within-question AUROC from 0.614 to 0.625 and mean cross-question C-Index from 0.738 to 0.757. Both averages change little beyond this competitor rank.

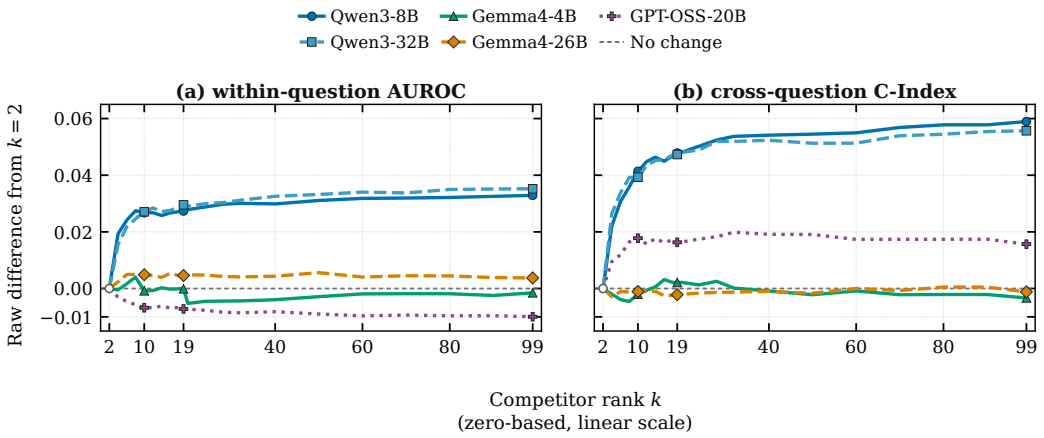


Figure 7: Rank sensitivity varies by model and target. Panels show each model's change in within-question AUROC (left) or cross-question C-Index (right) relative to $k = 2$. Curves show averages across three datasets at each of the 21 evaluated values of k; markers highlight $k \in \{2, 10, 19, 99\}$. Table 8 reports the corresponding absolute discrimination scores.

Table 8: Absolute logit-gap discrimination at selected competitor ranks. Cells report mean within-question AUROC (W) and cross-question C-Index (C) over three datasets per model; Overall averages all 15 settings. Light blue identifies $k = 19$, corresponding to the 20th most probable token used in the main comparison. Bold marks the highest W and C separately within each column across displayed indices, including ties at three decimal places. The response collection differs from that used in the main four-metric comparison in Figure 2.

Index k	**Per-model average (W/C)**					**Overall**
	Qwen3-8B	Qwen3-32B	Gemma4-4B	Gemma4-26B	GPT-OSS-20B	W / C
2	0.635 / 0.745	0.642 / 0.721	0.699 / 0.751	0.590 / **0.790**	**0.506** / 0.685	0.614 / 0.738
6	0.659 / 0.775	0.664 / 0.755	**0.701** / 0.747	**0.595** / 0.789	0.502 / 0.697	0.624 / 0.753
10	0.661 / 0.786	0.669 / 0.760	0.698 / 0.749	**0.595** / 0.789	0.500 / 0.703	0.625 / 0.757
14	0.660 / 0.791	0.669 / 0.766	0.699 / 0.752	0.594 / 0.789	0.500 / 0.703	0.625 / 0.760
18	0.662 / 0.791	0.671 / 0.768	0.699 / **0.753**	0.594 / 0.788	0.500 / 0.702	0.625 / 0.761
19	0.662 / 0.792	0.672 / 0.768	0.699 / **0.753**	0.594 / 0.788	0.499 / 0.702	0.625 / 0.761
20	0.663 / 0.793	0.671 / 0.769	0.694 / **0.753**	**0.595** / 0.788	0.499 / 0.702	0.624 / 0.761
40	0.665 / 0.799	0.675 / 0.773	0.695 / 0.750	0.594 / 0.789	0.498 / **0.704**	0.625 / 0.763
60	0.666 / 0.799	0.676 / 0.772	0.697 / 0.750	0.594 / **0.790**	0.497 / 0.703	**0.626** / 0.763
80	0.667 / 0.802	**0.677** / 0.775	0.697 / 0.749	0.594 / **0.790**	0.497 / 0.703	**0.626** / **0.764**
99	**0.668** / **0.803**	**0.677** / **0.777**	0.697 / 0.748	0.594 / 0.789	0.496 / 0.701	**0.626** / **0.764**

B Details for Section 4.2

B.1 Token Categories and Positions Within Words

We compute logit gap Δ_{19}, self-certainty, negative entropy, and maximum log probability using the definitions in Section 3.2 and Equation 1, with $K = 20$. The local diagnostics use all responses sampled across the 15 combinations of models and datasets.

We assign tokens to five categories using their decoded characters. Structural comprises empty strings and tokens containing only whitespace, markup, or punctuation. Among the remaining strings, those containing digits are labeled Digit. Mathematical or operator symbols are labeled Symbolic. Other alphabetic strings are Semantic; residual cases are Others. These categories are defined by the decoded characters and do not specify the role that a token plays in the reasoning process.

We group tokens into alphabetic words using heuristic boundaries based on the decoded characters. Each token within a word is a subword, or piece. We analyze two-, three-, and four-piece words

separately, with each word occurrence contributing one observation at every position. Table 9 reports the observed coverage for the token categories and word-length groups.

Table 9: Coverage by token category and word length for all sampled responses (five models, three datasets). Occurrences count tokens by category and words by position group. Symbolic excludes digits; Digit contains digits. Traces and model–question pairs can belong to multiple groups; the coverage is shared by all subword positions within each word-length group.

Group	Occurrences	Traces	Model–questions	Settings
Semantic	2,758,980,303	660,480	1,290	15/15
Structural	1,535,677,651	660,480	1,290	15/15
Symbolic	1,311,805,321	660,480	1,290	15/15
Digit	994,217,932	652,510	1,288	15/15
Others	25,618,508	328,951	873	11/15
2-piece words	136,824,127	655,934	1,290	15/15
3-piece words	34,707,433	554,108	1,279	15/15
4-piece words	11,382,863	340,353	1,075	15/15

Within each setting, we average token scores by category and compare each category mean with the Semantic mean. For all four metrics, Structural, Symbolic, and Digit tokens have higher mean certainty than Semantic tokens in every setting. For each metric, the 95% paired question-bootstrap intervals for all 45 category contrasts are above zero. Table 11 reports the raw means.

Table 10 compares the all-token mean of Δ_{19} (Original) with an equal-weight average of the Semantic, Structural, Symbolic, and Digit means (Equal), excluding Others from Equal. For cross-question scores, we pool tokens across responses to each question. Both use matched samples under the eligibility rules in Appendix E.2. For most models, Equal improves discrimination on both targets.

Table 10: Equal category weighting improves discrimination on both targets for most models. Original averages all token scores; Equal assigns 25% to each of the Semantic, Structural, Symbolic, and Digit means, excluding Others. Both use logit gap Δ_{19} with the scoring and sample-selection rules in Appendix B.1. Entries are equally weighted averages across the three datasets. Bold marks the higher discrimination value for each model and prediction target, including ties.

Model	**cross-question** C-Index ↑		**within-question** AUROC ↑	
	Original	Equal	Original	Equal
Qwen3-8B	0.7673	**0.8070**	0.6739	**0.6842**
Qwen3-32B	0.7720	**0.7926**	0.6467	**0.6552**
Gemma 4-E4B	0.7717	**0.7962**	**0.6556**	0.6481
Gemma4-26B	0.7236	**0.7240**	0.5675	**0.5907**
GPT-OSS-20B	0.6546	**0.6819**	0.5307	**0.5498**

For word positions, we average scores by word length and position within each setting and compare each later position with the first. Later pieces have higher mean certainty in all 90 comparisons per metric: six later positions across the three word lengths in 15 settings. All paired intervals are above zero for negative entropy and maximum log probability. Logit gap and self-certainty each have two intervals crossing zero, all involving Gemma4-26B. Certainty need not increase monotonically with subword position. These associations are consistent with constraints on word completion, although tokenization and word context are not separately controlled. Table 12 reports the position means; Appendix E.2 describes how the intervals are computed.

Table 11: Token-category means for four certainty metrics. For each model shown, Structural, Symbolic, and Digit means exceed the Semantic mean. Means use all sampled responses, with equal weight for token occurrences within each setting. Model rows average three datasets equally; Overall averages 15 settings. Symbolic excludes digits; Digit contains digits. NA denotes a missing category in at least one required setting. Darker shading denotes larger raw means; the color scale used for each metric here is the same as the scale for that metric in Table 12.

Model	Token category				
	Semantic	Structural	Symbolic	Digit	Others
A. Logit gap (Δ_{19})			Color scale: 8.159		29.115
Qwen3-8B	16.794	21.585	22.441	27.802	25.962
Qwen3-32B	17.439	22.178	22.772	26.863	24.600
Gemma 4-E4B	12.946	17.217	18.775	21.679	NA
Gemma4-26B	16.147	18.811	19.616	21.501	NA
GPT-OSS-20B	9.745	12.647	14.591	17.469	17.174
Overall	14.614	18.488	19.639	23.063	NA
B. Self-certainty			Color scale: 6.308		24.253
Qwen3-8B	13.025	17.720	18.672	23.007	21.614
Qwen3-32B	13.417	18.124	18.863	22.737	20.398
Gemma 4-E4B	10.565	14.415	15.807	18.507	NA
Gemma4-26B	13.660	15.987	16.545	18.424	NA
GPT-OSS-20B	7.764	10.292	11.962	14.669	14.082
Overall	11.686	15.308	16.370	19.469	NA
C. Negative entropy			Color scale: −1.2735		−0.0029
Qwen3-8B	−0.4643	−0.2181	−0.1465	−0.0295	−0.0884
Qwen3-32B	−0.4767	−0.2315	−0.1585	−0.0278	−0.0911
Gemma 4-E4B	−0.5726	−0.2066	−0.1269	−0.0412	NA
Gemma4-26B	−0.2721	−0.1062	−0.0827	−0.0261	NA
GPT-OSS-20B	−0.8892	−0.4257	−0.3013	−0.1090	−0.2812
Overall	−0.5350	−0.2376	−0.1632	−0.0467	NA
D. Maximum log probability			Color scale: −0.6874		−0.0011
Qwen3-8B	−0.2300	−0.1076	−0.0718	−0.0143	−0.0431
Qwen3-32B	−0.2396	−0.1165	−0.0788	−0.0136	−0.0446
Gemma 4-E4B	−0.2811	−0.0938	−0.0593	−0.0191	NA
Gemma4-26B	−0.1335	−0.0492	−0.0391	−0.0122	NA
GPT-OSS-20B	−0.4341	−0.1933	−0.1383	−0.0506	−0.1354
Overall	−0.2636	−0.1121	−0.0775	−0.0220	NA

B.2 Can Simple Token Masks Improve Correctness Ranking?

We keep the generated traces fixed and compare token masks that select the positions used to compute mean certainty. All-token (All in the tables) uses every position. First-piece retains only the first piece of each detected multi-token word or number and leaves other tokens unchanged. Exclude Digit and Others removes tokens assigned to either category. For each mask, we compute mean certainty by averaging the scores at the token positions retained under that condition.

We use the same eligible responses for all metrics and masks (Appendix E.2). We evaluate within-question AUROC and cross-question C-Index as defined in Section 3.1 and report paired changes from All-token by model and dataset (Tables 13 and 14).

Across the four metrics, First-piece improves within-question AUROC in only 6–8 of 15 settings, and excluding Digit and Others improves it in 8–9. Overall gains range from 0.0013 to 0.0031 and

Table 12: Later subwords have higher mean certainty than the first across metrics and models, although certainty need not increase monotonically with position. Cells report raw means over word occurrences in all sampled responses, separately for two-, three-, and four-piece words. Pos. denotes the position of a subword within a word. Model rows average three datasets equally; Overall averages 15 settings. Darker shading denotes larger raw means; each metric shares a color scale with Table 11.

Model	2-piece words		3-piece words			4-piece words			
	Pos. 1	Pos. 2	Pos. 1	Pos. 2	Pos. 3	Pos. 1	Pos. 2	Pos. 3	Pos. 4
A. Logit gap (Δ_{19})						Color scale: 8.159			29.115
Qwen3-8B	17.231	25.294	16.626	24.103	26.091	17.233	25.640	25.841	26.407
Qwen3-32B	17.471	23.601	16.301	22.008	23.497	16.878	22.479	22.248	22.988
Gemma 4-E4B	12.247	18.966	11.141	18.288	20.339	10.886	17.691	19.377	19.690
Gemma4-26B	17.397	21.277	17.512	20.619	22.103	17.116	19.651	20.999	21.404
GPT-OSS-20B	11.042	17.717	9.367	16.900	18.560	10.194	16.576	17.952	18.934
Overall	15.078	21.371	14.189	20.384	22.118	14.461	20.407	21.284	21.885
B. Self-certainty						Color scale: 6.308			24.253
Qwen3-8B	13.688	21.466	13.038	20.440	21.990	13.559	21.872	22.097	22.335
Qwen3-32B	13.726	20.026	12.595	18.671	19.850	13.139	19.139	18.999	19.414
Gemma 4-E4B	9.936	15.887	8.999	15.098	17.065	8.666	14.456	15.834	16.307
Gemma4-26B	14.707	18.428	14.825	17.827	19.130	14.066	16.712	17.917	18.513
GPT-OSS-20B	8.903	14.973	7.459	14.209	15.689	8.135	14.081	15.386	16.096
Overall	12.192	18.156	11.383	17.249	18.745	11.513	17.252	18.046	18.533
C. Negative entropy						Color scale: −1.2735			−0.0029
Qwen3-8B	−0.3538	−0.0316	−0.4088	−0.0290	−0.0295	−0.3857	−0.0089	−0.0167	−0.0187
Qwen3-32B	−0.3700	−0.0355	−0.4502	−0.0310	−0.0323	−0.4117	−0.0116	−0.0167	−0.0232
Gemma 4-E4B	−0.6146	−0.0795	−0.7593	−0.1840	−0.0337	−0.8977	−0.1160	−0.0786	−0.1512
Gemma4-26B	−0.2376	−0.0249	−0.2188	−0.0267	−0.0085	−0.3147	−0.0954	−0.0545	−0.0445
GPT-OSS-20B	−0.6606	−0.0800	−0.8863	−0.1430	−0.0641	−0.6970	−0.0626	−0.0521	−0.0523
Overall	−0.4473	−0.0503	−0.5447	−0.0827	−0.0336	−0.5414	−0.0589	−0.0437	−0.0580
D. Maximum log probability						Color scale: −0.6874			−0.0011
Qwen3-8B	−0.1715	−0.0149	−0.1979	−0.0145	−0.0143	−0.1902	−0.0040	−0.0079	−0.0087
Qwen3-32B	−0.1819	−0.0170	−0.2201	−0.0153	−0.0155	−0.2019	−0.0053	−0.0080	−0.0111
Gemma 4-E4B	−0.3088	−0.0360	−0.3722	−0.0914	−0.0148	−0.4731	−0.0490	−0.0354	−0.0729
Gemma4-26B	−0.1159	−0.0119	−0.1060	−0.0146	−0.0038	−0.1556	−0.0402	−0.0218	−0.0236
GPT-OSS-20B	−0.3184	−0.0342	−0.4342	−0.0670	−0.0271	−0.3446	−0.0271	−0.0218	−0.0233
Overall	−0.2193	−0.0228	−0.2661	−0.0406	−0.0151	−0.2731	−0.0251	−0.0190	−0.0279

from 0.0054 to 0.0083, respectively. First-piece improves within-question AUROC on both AIME datasets but reduces it on GPQA-Diamond for every metric. Cross-question effects also vary: for Gemma4-26B, excluding Digit and Others reduces the cross-question C-Index of maximum log probability by 0.0604 (95% paired question-bootstrap interval: $[-0.1263, -0.0088]$).

Table 13: Masking effects by model. Overall gains are small, and effects vary with the metric, model, and prediction target. Model rows average the three datasets equally; Overall averages all 15 settings. All denotes the all-token baseline; Δ is the paired change from All. Brackets show 95% paired question-bootstrap intervals. Bold marks the best condition within each row and target, including ties. Appendices B.2 and E.2 describe the paired bootstrap procedure used to obtain the intervals and define the token masks used in each evaluation condition.

Model	within-question AUROC			cross-question C-Index		
	All	Δ First-piece	Δ Exclude Digit and Others	All	Δ First-piece	Δ Exclude Digit and Others
A. Logit gap (Δ_{19})						
Qwen3-8B	0.674	+0.0024 [−0.0030,0.0086]	**+0.0074** [−0.0005,0.0172]	0.764	+0.0175 [0.0010,0.0400]	**+0.0294** [0.0102,0.0539]
Qwen3-32B	**0.647**	−0.0020 [−0.0078,0.0035]	−0.0033 [−0.0133,0.0056]	0.758	**+0.0024** [−0.0161,0.0206]	−0.0023 [−0.0319,0.0248]
Gemma 4-E4B	0.656	**+0.0015** [−0.0028,0.0063]	+0.0003 [−0.0095,0.0102]	0.776	+0.0034 [−0.0096,0.0181]	**+0.0123** [−0.0022,0.0286]
Gemma4-26B	0.568	+0.0110 [0.0020,0.0207]	**+0.0219** [−0.0019,0.0465]	**0.732**	−0.0003 [−0.0217,0.0209]	−0.0106 [−0.0422,0.0139]
GPT-OSS-20B	0.529	+0.0027 [−0.0021,0.0077]	**+0.0151** [0.0039,0.0274]	0.668	−0.0003 [−0.0140,0.0111]	**+0.0230** [0.0010,0.0494]
Overall	0.615	+0.0031 [0.0000,0.0064]	**+0.0083** [0.0012,0.0154]	0.740	+0.0045 [−0.0032,0.0135]	**+0.0104** [−0.0013,0.0229]
B. Self-certainty						
Qwen3-8B	0.672	+0.0013 [−0.0046,0.0078]	**+0.0048** [−0.0045,0.0150]	0.752	+0.0218 [0.0026,0.0454]	**+0.0330** [0.0097,0.0614]
Qwen3-32B	**0.644**	−0.0035 [−0.0099,0.0025]	−0.0054 [−0.0168,0.0047]	0.750	+0.0032 [−0.0170,0.0248]	**+0.0061** [−0.0229,0.0377]
Gemma 4-E4B	**0.655**	−0.0006 [−0.0078,0.0056]	−0.0020 [−0.0117,0.0070]	0.775	+0.0087 [−0.0058,0.0278]	**+0.0125** [−0.0039,0.0323]
Gemma4-26B	0.573	+0.0087 [−0.0008,0.0198]	**+0.0150** [−0.0085,0.0407]	**0.728**	−0.0048 [−0.0303,0.0170]	−0.0100 [−0.0425,0.0167]
GPT-OSS-20B	0.529	+0.0021 [−0.0031,0.0076]	**+0.0146** [0.0030,0.0275]	0.662	+0.0019 [−0.0086,0.0129]	**+0.0255** [0.0051,0.0507]
Overall	0.615	+0.0016 [−0.0020,0.0053]	**+0.0054** [−0.0022,0.0128]	0.733	+0.0062 [−0.0028,0.0164]	**+0.0134** [0.0007,0.0274]
C. Negative entropy						
Qwen3-8B	0.499	+0.0033 [−0.0053,0.0132]	**+0.0095** [−0.0054,0.0256]	0.647	**+0.0270** [−0.0033,0.0567]	+0.0240 [−0.0068,0.0620]
Qwen3-32B	0.502	+0.0014 [−0.0045,0.0077]	**+0.0085** [−0.0049,0.0237]	0.643	+0.0182 [−0.0075,0.0478]	**+0.0252** [−0.0070,0.0618]
Gemma 4-E4B	0.666	**+0.0002** [−0.0040,0.0048]	−0.0002 [−0.0071,0.0067]	0.774	+0.0021 [−0.0065,0.0110]	**+0.0054** [−0.0070,0.0187]
Gemma4-26B	0.482	+0.0009 [−0.0072,0.0091]	**+0.0097** [−0.0108,0.0319]	**0.623**	−0.0226 [−0.0636,0.0077]	−0.0541 [−0.1158,−0.0053]
GPT-OSS-20B	0.510	+0.0007 [−0.0029,0.0041]	**+0.0116** [0.0024,0.0203]	0.632	+0.0064 [−0.0033,0.0172]	**+0.0152** [−0.0038,0.0344]
Overall	0.532	+0.0013 [−0.0021,0.0047]	**+0.0078** [0.0009,0.0153]	0.664	**+0.0062** [−0.0068,0.0190]	+0.0031 [−0.0143,0.0218]
D. Maximum log probability						
Qwen3-8B	0.493	+0.0033 [−0.0055,0.0128]	**+0.0093** [−0.0054,0.0256]	0.640	**+0.0260** [−0.0034,0.0561]	+0.0250 [−0.0079,0.0641]
Qwen3-32B	0.501	+0.0013 [−0.0045,0.0074]	**+0.0077** [−0.0054,0.0224]	0.642	+0.0146 [−0.0096,0.0411]	**+0.0212** [−0.0130,0.0595]
Gemma 4-E4B	0.667	**+0.0010** [−0.0032,0.0055]	−0.0008 [−0.0087,0.0064]	0.775	+0.0016 [−0.0087,0.0113]	**+0.0060** [−0.0061,0.0195]
Gemma4-26B	0.471	+0.0038 [−0.0055,0.0138]	**+0.0086** [−0.0115,0.0284]	**0.610**	−0.0325 [−0.0693,−0.0032]	−0.0604 [−0.1263,−0.0088]
GPT-OSS-20B	0.508	+0.0001 [−0.0036,0.0038]	**+0.0128** [0.0044,0.0209]	0.628	+0.0052 [−0.0046,0.0148]	**+0.0102** [−0.0055,0.0277]
Overall	0.528	+0.0019 [−0.0015,0.0056]	**+0.0075** [0.0006,0.0147]	0.659	**+0.0030** [−0.0088,0.0151]	+0.0004 [−0.0171,0.0197]

Table 14: Masking effects by dataset. First-piece improves within-question AUROC on both AIME datasets but reduces it on GPQA-Diamond for all four metrics. Rows average the five models equally. All denotes the all-token baseline; Δ is the paired change from All. Brackets show 95% paired question-bootstrap intervals. Bold marks the best condition within each row and target, including ties. Appendices B.2 and E.2 provide the token-mask definitions and describe how paired bootstrap resampling of questions is used to obtain the reported intervals.

Dataset	within-question AUROC			cross-question C-Index		
	All	Δ First-piece	Δ Exclude Digit and Others	All	Δ First-piece	Δ Exclude Digit and Others
A. Logit gap (Δ_{19})						
AIME 2024	0.624	+0.0075 [0.0015,0.0139]	**+0.0247** [0.0076,0.0426]	0.759	+0.0044 [−0.0106,0.0228]	**+0.0161** [−0.0121,0.0461]
AIME 2025	0.637	**+0.0052** [−0.0006,0.0116]	+0.0006 [−0.0096,0.0109]	0.783	**+0.0058** [−0.0104,0.0252]	+0.0057 [−0.0147,0.0275]
GPQA-Diamond	**0.583**	−0.0034 [−0.0072,0.0002]	−0.0004 [−0.0040,0.0037]	0.677	+0.0034 [−0.0029,0.0101]	**+0.0092** [0.0017,0.0166]
B. Self-certainty						
AIME 2024	0.623	+0.0070 [−0.0008,0.0146]	**+0.0216** [0.0027,0.0416]	0.749	+0.0138 [−0.0047,0.0371]	**+0.0250** [−0.0060,0.0605]
AIME 2025	0.638	**+0.0023** [−0.0042,0.0087]	−0.0049 [−0.0167,0.0052]	0.778	+0.0006 [−0.0191,0.0221]	**+0.0056** [−0.0183,0.0314]
GPQA-Diamond	**0.583**	−0.0044 [−0.0083,−0.0005]	−0.0004 [−0.0040,0.0035]	0.673	+0.0041 [−0.0027,0.0108]	**+0.0096** [0.0016,0.0177]
C. Negative entropy						
AIME 2024	0.522	+0.0054 [−0.0005,0.0120]	**+0.0257** [0.0093,0.0437]	0.692	**+0.0073** [−0.0208,0.0372]	−0.0131 [−0.0534,0.0301]
AIME 2025	0.526	**+0.0029** [−0.0045,0.0108]	−0.0011 [−0.0140,0.0120]	0.657	+0.0021 [−0.0244,0.0267]	**+0.0159** [−0.0186,0.0535]
GPQA-Diamond	**0.547**	−0.0044 [−0.0076,−0.0013]	−0.0012 [−0.0040,0.0018]	0.641	**+0.0092** [0.0030,0.0162]	+0.0066 [−0.0021,0.0151]
D. Maximum log probability						
AIME 2024	0.518	+0.0077 [0.0012,0.0156]	**+0.0264** [0.0099,0.0443]	0.685	**+0.0019** [−0.0228,0.0287]	−0.0164 [−0.0575,0.0291]
AIME 2025	0.520	**+0.0027** [−0.0041,0.0104]	−0.0031 [−0.0163,0.0102]	0.652	−0.0025 [−0.0291,0.0218]	**+0.0114** [−0.0230,0.0475]
GPQA-Diamond	**0.546**	−0.0047 [−0.0077,−0.0018]	−0.0008 [−0.0037,0.0023]	0.639	**+0.0095** [0.0030,0.0166]	+0.0061 [−0.0027,0.0150]

C Details for Section 5

Table 15 compares model-level results for 1,000-token prefixes and suffixes with full-trace scores.

C.1 Sensitivity to Window Length

Section 5 uses matched 1,000-token windows to compare where evidence for the two prediction targets becomes available. Here we test whether the results depend on window length, the summary and final-answer tokens included in late windows, or the choice of certainty metric. We retain the main text's cross-question C-Index and within-question AUROC; the window-length comparisons focus on prefixes for cross-question difficulty and suffixes for within-question trace correctness.

We use a fixed 1,000-token window across models and datasets. Table 16 shows that early prefixes remain informative about cross-question difficulty and short suffixes retain higher within-question AUROC than the 30,000-token reference across neighboring window lengths. From 0.5k to 2k tokens, prefix cross-question C-Index ranges from 0.691 to 0.735, while suffix within-question AUROC ranges from 0.624 to 0.638; every suffix value exceeds the 0.611 reference obtained from the final 30,000 tokens. The cross-question C-Index generally increases as the prefix expands. The within-question AUROC is strongest for shorter suffixes and generally decreases as the window expands to include earlier trace regions. Discrimination varies across models and datasets.

Table 15: Temporal localization across models. Both prediction targets use the same three aggregation regions: the first 1,000 tokens, the final 1,000 tokens, and the full trace. The cross-question C-Index ranks empirical pass rates across questions, whereas within-question AUROC ranks correct and incorrect traces for the same question. Each model row equally averages AIME 2024, AIME 2025, and GPQA-Diamond; Overall equally averages all model–dataset combinations. Higher is better. Blue shading marks the windows selected for each target (cross-question prefix and within-question suffix); it does not indicate which aggregation region has the highest score in each row.

Model	cross-question C-Index			within-question AUROC		
	Prefix (1k)	Suffix (1k)	Full Trace	Prefix (1k)	**Suffix (1k)**	Full Trace
Qwen3-8B	**0.709**	0.774	0.764	0.505	**0.683**	0.674
Qwen3-32B	**0.728**	0.769	0.758	0.502	**0.652**	0.647
Gemma4-4B	**0.721**	0.748	0.776	0.489	**0.675**	0.656
Gemma4-26B	**0.753**	0.701	0.732	0.514	**0.533**	0.568
GPT-OSS-20B	**0.701**	0.664	0.668	0.491	**0.584**	0.529
Overall	**0.723**	**0.731**	**0.740**	**0.500**	**0.626**	**0.615**

Table 16: Effect of absolute window length on prefix cross-question C-Index and suffix within-question AUROC. Entries give equal-weight averages across the 15 model–dataset combinations, with the observed minimum–maximum range in brackets. Windows count generated tokens. The 30,000-token suffix approximates the full trace, covering 98.4% of responses. Higher is better.

(a) 100–500 generated tokens

Readout	100	200	300	400	500
cross-question: prefix C-Index	0.578 [0.449, 0.664]	0.642 [0.492, 0.731]	0.670 [0.550, 0.744]	0.682 [0.595, 0.754]	0.691 [0.608, 0.754]
within-question: suffix AUROC	0.639 [0.501, 0.742]	0.639 [0.402, 0.756]	0.640 [0.443, 0.758]	0.633 [0.414, 0.771]	0.638 [0.429, 0.791]

(b) 1,000–5,000 generated tokens

Readout	1,000	2,000	3,000	4,000	5,000
cross-question: prefix C-Index	0.723 [0.637, 0.808]	0.735 [0.649, 0.841]	0.737 [0.652, 0.857]	0.731 [0.651, 0.841]	0.726 [0.649, 0.846]
within-question: suffix AUROC	0.626 [0.509, 0.737]	0.624 [0.448, 0.738]	0.610 [0.376, 0.732]	0.611 [0.408, 0.740]	0.606 [0.411, 0.721]

(c) 10,000–30,000 generated tokens

Readout	10,000	15,000	20,000	30,000
cross-question: prefix C-Index	0.720 [0.642, 0.870]	0.731 [0.645, 0.875]	0.740 [0.645, 0.875]	0.742 [0.638, 0.875]
within-question: suffix AUROC	0.609 [0.469, 0.713]	0.611 [0.461, 0.713]	0.611 [0.461, 0.705]	0.611 [0.480, 0.706]

Table 17 repeats the sweep using fractions of each trace. Prefix cross-question performance again generally rises as more context is included. Suffix within-question AUROC instead peaks at 0.651 for the final 5% and falls to 0.616–0.619 over the final 20%–70%. The broad temporal pattern also appears with relative windows, although discrimination varies across settings and window lengths.

C.2 Effects of Summary and Answer Content

Late windows overlap summaries and final answers, so their higher within-question scores could reflect which parts of a response they contain. We therefore compare the suffix before and after removing final-answer spans and compare it with the full summary. We also divide the summary into four consecutive parts, labeled Q1–Q4 from first to last, each containing approximately one quarter of the summary tokens. Table 18 reports the absolute scores under these controls.

Table 17: Effect of relative window length on prefix cross-question C-Index and suffix within-question AUROC. Entries average the 15 model–dataset combinations equally, with the observed minimum–maximum range in brackets. Windows are fractions of each trace. Higher is better.

(a) 0.1%–5% of the trace

Readout	0.1%	0.5%	1%	2%	5%
cross-question: prefix C-Index	0.508 [0.210, 0.851]	0.627 [0.493, 0.831]	0.672 [0.542, 0.794]	0.693 [0.565, 0.797]	0.688 [0.605, 0.789]
within-question: suffix AUROC	0.611 [0.447, 0.780]	0.621 [0.457, 0.753]	0.628 [0.497, 0.736]	0.636 [0.501, 0.755]	0.651 [0.567, 0.743]

(b) 10%–70% of the trace

Readout	10%	20%	30%	50%	70%
cross-question: prefix C-Index	0.697 [0.590, 0.819]	0.708 [0.564, 0.852]	0.715 [0.564, 0.874]	0.719 [0.533, 0.846]	0.724 [0.540, 0.873]
within-question: suffix AUROC	0.636 [0.505, 0.745]	0.617 [0.445, 0.720]	0.616 [0.469, 0.718]	0.619 [0.490, 0.715]	0.618 [0.479, 0.713]

Table 18: Scores for the final 1,000-token suffix before and after removing final-answer spans, the full summary, and its four consecutive quarters (Q1–Q4). Quarters contain approximately equal token counts and are ordered from first to last. Values average the 15 model–dataset settings equally; higher values indicate better discrimination for the corresponding prediction target.

Control dimension	Aggregation region	within-question AUROC	cross-question C-Index
Answer-span content	Suffix (1k)	0.6260	0.7310
	Suffix (1k), answers removed	0.6279	0.7334
Summary extent	Full summary	0.6100	0.7414
	Suffix (1k)	0.6260	0.7310
Within-summary position	Summary Q1	0.5371	0.6950
	Summary Q2	0.5353	0.7021
	Summary Q3	0.5684	0.7090
	Summary Q4	0.6208	0.6822

Answer removal leaves both aggregate scores nearly unchanged. The within-question AUROC is higher for the suffix than for the full summary (0.6260 versus 0.6100) and rises from 0.5371 in summary Q1 to 0.6208 in Q4. The cross-question scores do not show the same increase toward the end. Summary content changes from Q1 to Q4. These comparisons do not isolate the effect of position from changes in content.

C.3 Comparisons across Certainty Metrics

Finally, we check whether the early cross-question signal and the higher within-question scores in suffixes also appear with metrics other than logit gap. In Table 19, we apply the same full-trace and 1,000-token comparisons to all four metrics defined in Sections 2 and 3.

Across all four metrics, suffixes outperform full traces in average within-question AUROC. Prefix cross-question scores retain most full-trace performance for logit gap and self-certainty and improve it for negative entropy and maximum log probability. The latter uses only the top token. Results vary across settings. Section 6 evaluates allocation from early scores and voting from late scores.

Table 19: Full-trace, prefix, and suffix scores across four certainty metrics. Both panels report equal-weight averages over the same 15 model–dataset combinations. For each metric, panel (a) compares the cross-question C-Index of the full trace with that of the 1,000-token prefix, and panel (b) compares the within-question AUROC of the full trace with that of the 1,000-token suffix. Each difference is the prefix or suffix score minus the full-trace score; the final column counts combinations on which that prefix or suffix score is higher. No exact ties occur. Blue shading identifies the prefix in panel (a) and the suffix in panel (b) and does not denote the largest value in a row. Higher is better.

(a) cross-question C-Index

Metric	Full trace	**Prefix (1k)**	Prefix (1k) − full trace	Combinations with prefix > full trace
Logit gap	0.740	**0.723**	−0.017	8/15
Self-certainty	0.733	**0.717**	−0.016	7/15
Negative entropy	0.664	**0.708**	+0.045	12/15
Maximum log probability	0.659	**0.709**	+0.050	12/15

(b) within-question AUROC

Metric	Full trace	**Suffix (1k)**	Suffix (1k) − full trace	Combinations with suffix > full trace
Logit gap	0.615	**0.626**	+0.011	9/15
Self-certainty	0.615	**0.626**	+0.012	9/15
Negative entropy	0.532	**0.589**	+0.057	14/15
Maximum log probability	0.528	**0.585**	+0.057	13/15

D Details for Section 6

This appendix specifies the TACS configuration in Table 1 and reports its results by model and dataset. We then examine how TACS distributes samples across questions and how often sampling reaches the trace limit, before comparing it with fixed-budget and adaptive baselines.

D.1 Experimental Protocol

We evaluate five models on 30 questions from each AIME dataset and 198 from GPQA-Diamond. TACS uses the top-1–top-20 logit gap Δ_{19} (Equation 6), averaged over the first 1,000 tokens for allocation and the final 500 tokens for voting. For traces shorter than the scoring window, we use their full length. Appendix E.3 provides sampling and cost-accounting details.

We calibrate the model-specific allocation thresholds on BRUMO 2025 (Balunović et al., 2025). The allocation thresholds θ_m are 19.0 for Qwen3-8B, 16.5 for Qwen3-32B, 14.0 for Gemma 4-E4B, 16.5 for Gemma4-26B, and 10.0 for GPT-OSS-20B. Each model's threshold is shared across datasets. We generate complete traces until 32 have prefix scores $\geq \theta_m$ or 256 have been sampled.

We use rank-weighted voting for observed responses, including those below the prefix threshold. For n voting responses, let r_i be response i's ascending certainty rank from 1 to n, with tied scores assigned their average rank. Its vote weight is $w_i = r_i^p$, with rank exponent $p = 1$. The answer with the largest total rank weight wins. The fixed policy uses certainty scores and generated answers to stop and vote.

D.2 Detailed Results and Voting Comparisons

Table 20 reports the main comparison across 15 model–dataset settings. Within either sampling condition, all four voting methods observe the same traces, so their accuracy differences arise from answer aggregation. Under early allocation, Suffix 500 rank-weighted voting achieves the highest accuracy in all nine settings involving Qwen or Gemma 4-E4B. For Gemma4-26B and GPT-OSS-20B, the voting method with the highest accuracy varies across datasets.

Sampling changes the candidates. With Suffix 500 rank-weighted voting, early allocation raises Qwen3-32B's AIME 2025 accuracy from 80.08% to 81.38%, but lowers Gemma4-26B's from

96.67% to 94.92%. All relative token costs are below one, ranging from 0.125 to 0.312. Overall accuracy is 79.54% at relative cost 0.176; the same voting rule yields 79.71% when applied to responses obtained with fixed 256-trace sampling.

Table 20: Voting by model and dataset. Majority voting gives one vote per response; other columns use rank-weighted voting ($p = 1$) with mean logit gap over the indicated region. Methods within each panel share traces and token cost. Overall averages the 15 settings equally; relative cost is each setting's total tokens divided by its fixed-256 total. Within each panel, bold marks the highest accuracy in each row, including all ties at the displayed precision.

(a) Fixed-budget sampling: 256 traces

		Accuracy (%) ↑				Relative token cost ↓
Model	**Dataset**	Majority voting	Full trace	Final 1,000 tokens	Final 500 tokens	
Qwen3-8B	AIME 2024	80.00	**83.02**	82.94	82.76	1.000
	AIME 2025	79.77	79.35	79.61	**79.79**	1.000
	GPQA-Diamond	66.52	66.07	66.54	**66.81**	1.000
Qwen3-32B	AIME 2024	86.02	90.55	88.80	**90.60**	1.000
	AIME 2025	80.00	80.05	**80.08**	**80.08**	1.000
	GPQA-Diamond	72.87	72.72	**73.45**	73.25	1.000
Gemma 4-E4B	AIME 2024	64.38	66.88	67.21	**67.94**	1.000
	AIME 2025	56.61	58.13	58.05	**59.01**	1.000
	GPQA-Diamond	65.35	66.15	65.67	**66.19**	1.000
Gemma4-26B	AIME 2024	**93.33**	**93.33**	**93.33**	**93.33**	1.000
	AIME 2025	96.59	**96.67**	96.61	**96.67**	1.000
	GPQA-Diamond	84.24	**84.54**	84.15	84.47	1.000
GPT-OSS-20B	AIME 2024	**93.33**	**93.33**	**93.33**	**93.33**	1.000
	AIME 2025	**94.01**	93.33	93.46	93.41	1.000
	GPQA-Diamond	67.61	67.67	**68.10**	68.06	1.000
Overall		78.71	79.45	79.42	**79.71**	1.000

(b) Early allocation: target of 32 qualifying traces; at most 256 traces

		Accuracy (%) ↑				Relative token cost ↓
Model	**Dataset**	Majority voting	Full trace	Final 1,000 tokens	Final 500 tokens	
Qwen3-8B	AIME 2024	80.70	83.31	83.54	**83.62**	0.126
	AIME 2025	78.36	77.79	78.96	**79.14**	0.125
	GPQA-Diamond	65.80	66.10	66.14	**66.24**	0.174
Qwen3-32B	AIME 2024	88.26	89.45	89.38	**90.21**	0.125
	AIME 2025	81.17	81.30	81.12	**81.38**	0.125
	GPQA-Diamond	72.60	73.37	73.63	**73.65**	0.184
Gemma 4-E4B	AIME 2024	64.40	67.32	67.60	**67.68**	0.132
	AIME 2025	54.53	56.15	56.59	**58.26**	0.125
	GPQA-Diamond	64.74	65.29	65.10	**65.46**	0.300
Gemma4-26B	AIME 2024	**93.93**	93.88	93.44	93.44	0.135
	AIME 2025	94.77	**95.03**	94.53	94.92	0.128
	GPQA-Diamond	83.90	**84.49**	84.17	84.22	0.312
GPT-OSS-20B	AIME 2024	93.31	**93.33**	93.15	93.20	0.279
	AIME 2025	93.12	92.76	**93.54**	93.49	0.177
	GPQA-Diamond	66.94	67.34	67.91	**68.20**	0.186
Overall		78.44	79.13	79.25	**79.54**	0.176

D.3 Sampling Allocation and Stopping

A shared target of 32 qualifying traces produces different sample counts across questions. For Figure 8, we define difficulty as one minus empirical accuracy on 512 reference responses per model–question pair, counting invalid answers as incorrect. Within each model, we pool the datasets and divide questions into easy, medium, and hard groups by difficulty rank, keeping ties together. We normalize each group's mean trace count by the model-wide mean across all 258 questions, weighting questions equally. Figure 8 shows that the mean number of traces allocated to the hard group exceeds that allocated to the easy group for every evaluated model.

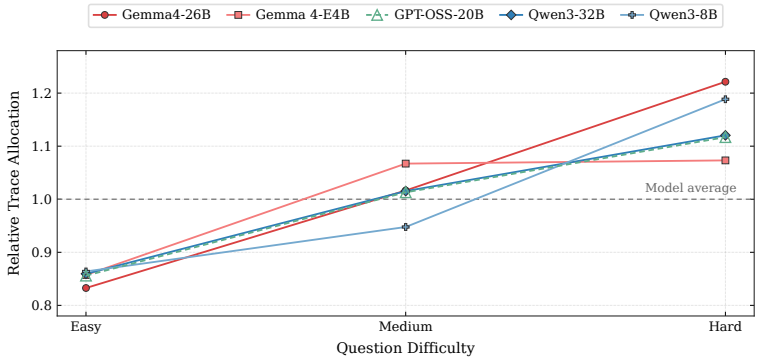


Figure 8: TACS allocates more traces to harder question groups. Group means are normalized by each model's mean across all 258 questions; the horizontal dashed line marks this reference.

Table 21 reports trace counts: every model has a median of 32, with means ranging from 41.14 to 71.50. About 13% of sampling runs reach the 256-trace limit for the Gemma models, compared with 1.94%–2.71% for the other models. Questions are weighted equally here; Table 23 weights datasets equally and reports a mean of 43.5 traces for TACS.

Table 21: Sample counts under early allocation, with a target of 32 qualifying traces and a limit of 256. Means and medians summarize trace counts per sampling run across each model's 258 questions, weighted equally. The final column gives the percentage of runs using 256 traces. Table 23 instead reports averages that assign equal weight to each dataset.

Model	Number of questions	Mean trace count	Median trace count	Sampling runs at the trace limit (%)
Qwen3-8B	258	41.14	32	1.94
Qwen3-32B	258	41.49	32	1.97
Gemma 4-E4B	258	71.50	32	12.82
Gemma4-26B	258	70.70	32	12.92
GPT-OSS-20B	258	43.24	32	2.71

D.4 Baseline Comparisons

Table 22 compares answer aggregation on the same 256 completed traces within each sampling run, at identical token cost. Self-Consistency (Wang et al., 2023) uses majority voting. Self-Certainty (Kang et al., 2025) uses certainty scores for Best-of-N selection or rank-weighted voting. Reasoning-Pruning Consistency combines pruning with probability-weighted voting using its released implementation[1]. Offline DeepConf (Fu et al., 2025) filters completed traces by confidence before applying confidence-weighted voting to the retained candidate responses.

Self-Certainty achieves 79.63% accuracy with rank-weighted voting, close to 79.71% for Suffix 500 rank-weighted voting, but 73.85% with Best-of-N. The contrast illustrates the importance of how

[1] https://github.com/WNJXYK/RPC

certainty scores are used. Suffix 500 rank-weighted voting has the highest overall point estimate among the listed methods and leads on AIME 2025 and GPQA-Diamond; DeepConf's tail-score variant, which retains 10% of the traces, leads on AIME 2024.

Table 22: Answer aggregation using the same 256 responses and token budget within each sampling run. Suffix 500 rank-weighted voting has the highest overall point estimate. Accuracy is in percent; differences are percentage points from Self-Consistency. Dataset columns average five models; Overall averages the 15 settings equally. Bold marks column maxima, including displayed ties. The rank exponents are 1.2 for Self-Certainty and 1 for Suffix 500 rank-weighted voting. DeepConf retains the indicated fraction of the highest-scoring traces. For DeepConf, bottom-group mean averages the lowest-scoring 10% of sliding windows, whereas tail mean takes the average of token confidence over the tokens in the final segment of each response.

Method	AIME 2024	AIME 2025	GPQA-Diamond	Overall	Difference (percentage points)
Self-Consistency	83.41	81.40	71.32	78.71	+0.00
Self-Certainty Best-of-N	80.94	71.53	69.07	73.85	-4.86
Self-Certainty: rank-weighted voting	85.78	81.58	71.53	79.63	+0.92
Reasoning-Pruning Consistency	84.47	80.53	70.72	78.57	-0.13
DeepConf: bottom-group mean; retain 10%	83.77	78.10	70.66	77.51	-1.20
DeepConf: bottom-group mean; retain 90%	84.16	81.44	71.35	78.98	+0.27
DeepConf: tail mean; retain 10%	**86.60**	79.58	71.68	79.29	+0.58
DeepConf: tail mean; retain 90%	84.47	81.38	71.48	79.11	+0.40
Suffix 500 rank-weighted voting	85.59	**81.79**	**71.76**	**79.71**	**+1.01**

We evaluate the adaptive baselines in Table 23 under matched sampling conditions. Adaptive-Consistency (Aggarwal et al., 2023) stops sampling based on a probabilistic criterion for answer consistency. Early-Stopping Self-Consistency (Li et al., 2024) stops when answers agree within a recent sampling window. Online DeepConf (Fu et al., 2025) uses confidence thresholds estimated from warmup traces to truncate low-confidence generations, then applies confidence-weighted voting over retained traces. DeepConf uses 2,048-token windows or tails; for online DeepConf, we calibrate thresholds using 16 complete warmup traces for each question.

Table 23: Accuracy and token cost under fixed and adaptive sampling. TACS reaches 79.54% accuracy at $0.176\times$ the fixed-256 Self-Consistency cost. All three measures are averaged equally across the 15 model–dataset settings. Early-Stopping Self-Consistency allows at most 256 traces on AIME and 255 on GPQA-Diamond. Retention parameters are 10% for DeepConf-low and 90% for DeepConf-high. DeepConf counts attempted traces and tokens from truncated generations; each individual trace observed by TACS is generated through to completion.

Method	**Sampling budget**	Accuracy (%)	Relative token cost	Mean trace count
Self-Consistency	32 traces (fixed)	78.27	0.125	32.0
Self-Consistency	128 traces (fixed)	78.63	0.500	128.0
Self-Consistency	256 traces (fixed)	78.71	1.000	256.0
Adaptive-Consistency	At most 256 traces	78.65	0.363	26.1
Early-Stopping Self-Consistency	Dataset-specific maximum	78.66	0.419	69.2
DeepConf-low	At most 256 traces	76.89	0.176	57.8
DeepConf-high	At most 256 traces	79.00	0.555	113.6
TACS	32 qualifying traces; at most 256 traces	79.54	0.176	43.5

TACS reaches 79.54% accuracy at relative token cost 0.176, compared with 78.65% at 0.363 for Adaptive-Consistency and 78.66% at 0.419 for Early-Stopping Self-Consistency. Both DeepConf variants have lower accuracy at equal or higher displayed token cost.

E Reproducibility Details

E.1 Prompt Templates

Prompts use a single user message and the model's chat template with a generation prompt. GPT-OSS-20B uses medium reasoning effort; Gemma models use thinking mode. Braces indicate placeholders.

AIME 2024 and AIME 2025

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

GPQA-Diamond

```
Answer the following multiple choice question.  Please reason step
by step, and put your final answer (one of A, B, C, D) within
\boxed{}.
{question}
A) {choice_A}
B) {choice_B}
C) {choice_C}
D) {choice_D}
```

E.2 Diagnostic Samples and Evaluation Eligibility

The four-metric comparison and local diagnostics use 512 responses per model–question pair across five models and three datasets, totaling 660,480 responses. Of these, 641,173 meet the evaluation eligibility criteria. The token-category and word-position means use all responses; correctness-based comparisons use the eligible subset. For these analyses, we extract the final boxed answer and retain responses with a nonempty answer, as well as any response graded correct. We refer to retained responses as valid; they include both correct and incorrect answers. We also repeat the local diagnostics using only valid responses. Table 9 in Appendix B.1 reports coverage of the all-response population used in the main local figures and tables. In the rank-sensitivity analysis, questions without valid responses and unestimable question statistics are excluded without imputation. In the masking comparisons, all conditions share the same responses, with no condition-specific exclusions of responses on the basis of missing or non-finite scores.

For category-weighting comparisons, both conditions use the same questions containing Digit tokens for cross-question evaluation and the same responses containing all four principal token categories for the evaluation of within-question discrimination between correct and incorrect responses.

For the comparisons in Appendices A.1–A.3, questions are resampled jointly across models, metrics, and comparisons, retaining all responses for each sampled question. We recompute all statistics and equal-weight averages in each replicate, using the original question identity to distinguish pairs from the same question and pairs from different questions.

E.3 Trace Sampling and TACS Simulation

The TACS experiments are offline simulations over pre-sampled trace pools. For each model–question pair, the main experiments use 1,024 stored responses and $R = 128$ shared replay orders generated with seed 42. Each order contains 256 distinct trace indices. We apply the stopping and voting rules specified in Appendix D.1 to each order: fixed-budget sampling uses all 256 responses, whereas early allocation observes an initial portion. Fixed-budget methods share the candidate responses and valid-answer rules within each replay. Adaptive baselines use the same stored pools and replay-order construction. TACS voting uses responses with valid answers and finite scores. Responses with invalid answers still count toward the 32-trace stopping target if their prefix scores qualify.

For the experiments in Section 6, token cost sums the complete generated lengths of all observed responses, including those with invalid answers. Within each setting, relative token cost divides this

total by the corresponding total for fixed 256-trace sampling on matched orders; reported averages weight these ratios equally, rather than averaging per-question token ratios. Reported costs describe completion tokens consumed by the replayed policies.

Table 24: Key decoding and inference settings used to generate the traces.

Parameter	Setting
Temperature	0.6
Top-p	0.95
Maximum generation length	32,768 tokens for Qwen models; 131,072 tokens for GPT-OSS and Gemma models
Batch size	256 traces per vLLM generation call; vLLM maximum concurrent sequences set to 512
Number of samples	1,024 stored responses per model–question pair for the main TACS simulations
Random seed	Offline resampling uses seed 42; trace generation does not pass an explicit vLLM sampling seed