Token-Level Verification under Controlled Evaluation: Protocol Sensitivity Shapes Apparent Performance

Yuhan Chi
yhchi25@m.fudan.edu.cn
Fudan University

Accepted as a poster at the ICML 2026 Workshop on AI for Math (AI4Math).
Code: https://github.com/Chi-Shan0707/token-verification-mirage

Abstract

Token-level uncertainty signals such as entropy, log-probabilities, and confidence scores are increasingly explored as lightweight aids for verifying LLM reasoning without additional model calls. Yet reported AUROC varies from near-chance (0.55) to high values (0.80+), with no consensus on what drives this variance. Prior work has not systematically isolated, within this math-verification setting, how much of this variation stems from evaluation protocol artifacts rather than genuine signal differences.

We conduct a controlled study across two benchmarks (MATH, BigMath), two model families (Qwen, Llama; three instances), and twelve analytical methods under within-problem leave-one-out (LOO) evaluation, GroupKFold cross-problem splits, and permutation-null calibration.

We find that evaluation protocol choices can shift AUROC more than the observed differences among several methods in our setting; switching from pooled to within-problem LOO evaluation shifts AUROC by 0.04–0.18. Under controlled evaluation, all twelve analyzed methods converge to an overlapping AUROC range of roughly 0.60–0.75 on hard math problems, with genuine excess above the permutation null (≈0.58–0.60) of only ≈0.05–0.15. Direction-agnostic (DA) scoring—selecting the better scoring direction per problem—yields this range, but a fixed-direction baseline (learning one global direction) reveals that final-token entropy collapses from 0.72–0.75 to 0.47–0.48 (below chance), indicating its apparent strength reflects per-problem direction selection rather than stable generalization. Two adapted prior token-level uncertainty baselines Wang et al. [2025], Malinin & Gales [2021] fall within the same range; a Kadavath-style P(True) self-evaluation baseline Kadavath et al. [2022] achieves higher AUROC than these shallow statistics under the same direction-agnostic evaluation. The accompanying repository provides the evaluation scripts and paper artifacts.

1 Introduction

In math-reasoning systems, *verification* asks whether sampled solutions can be ranked by correctness without additional expensive evaluation Hendrycks et al. [2021], Cobbe et al. [2021]. In full AI-for-math systems, this role is often served by stronger mechanisms such as symbolic checking, code or tool execution, learned verifiers, reward models, process supervision, or self-consistency. These approaches can be effective but may require additional model calls, labeled data, external tools, or fine-tuning. Lightweight alternatives extract verification signals directly from the token-level statistics already produced during generation—entropy, log-probabilities, and confidence trajectories—without any additional model calls Wang et al. [2025], Malinin & Gales [2021]. A related but richer

family of approaches prompts the model to evaluate its own answer, e.g. P(True) self-evaluation Kadavath et al. [2022].

This premise has motivated a growing line of work. Wang et al. Wang et al. [2025] use entropy-after-thinking signals for reasoning-model early exit; we adapt this entropy-trajectory idea as one lightweight uncertainty baseline. Malinin & Gales Malinin & Gales [2021] derive token-level uncertainty decompositions from ensemble distributions. Kadavath et al. Kadavath et al. [2022] show that models can self-assess answer correctness via P(True) prompts. Self-consistency reranking Wang et al. [2023] uses answer agreement across samples. Process reward models Lightman et al. [2023], Uesato et al. [2022] and neuron-agreement analyses Chen et al. [2025] access richer supervision or internal signals. These methods span a wide design space; our controlled study focuses on the shallow external token-statistic end of that space.

Reported results, however, are inconsistent. AUROC values for token-level verifiers span 0.55–0.80+ across studies, with no consensus on whether this variation reflects genuine methodological advances or evaluation artifacts. This makes it difficult to assess whether additional verification features improve over simple baselines.

Our experiments indicate that evaluation design is a major contributor to the observed variation. Global pooled evaluation—pooling predictions across problems—conflates problem-level difficulty with solution-level correctness, allowing a verifier to exploit problem-level difficulty signals rather than genuine per-solution discrimination, shifting AUROC by 0.04–0.18 relative to within-problem evaluation. In-sample evaluation of within-problem methods introduces subtle leakage: using a held-out trace's neighbors to construct scoring centroids biases AUROC estimates. These two protocol artifacts, taken together, can shift measured AUROC by 0.04–0.18 (Table 2), an effect that exceeds the entire range of genuine method-driven variation under controlled evaluation.

Instead of proposing a new verification method, we audit what shallow external token statistics can extract under controlled evaluation. This diagnostic focus is deliberate: before token-level signals are used as low-cost filters, auxiliary rerankers, or baselines for stronger verifiers, their evaluation protocol must not overstate their standalone discriminative power. We enforce three methodological controls: (i) within-problem LOO evaluation that eliminates problem-identity leakage; (ii) GroupK-Fold by problem for cross-problem analyses; and (iii) permutation-null calibration that accounts for direction-agnostic scoring inflation. Under this protocol, we benchmark twelve analytical methods across two model families and two benchmarks, and implement three prior-inspired baselines to check whether related uncertainty and self-evaluation signals fall in the same range.

In our experiments, protocol choice has a larger effect than feature design on reported performance. Under controlled evaluation, the analyzed methods occupy a narrow DA AUROC range of 0.60–0.75 on hard problems, with bootstrap CIs that substantially overlap. A key practical distinction emerges between oracle and fixed-direction evaluation: final-token entropy achieves the highest DA AUROC but collapses to chance (0.47–0.48) under fixed-direction AUROC, because the entropy-correctness direction varies idiosyncratically across problems. The prior-inspired baselines fall within or above the same range, with the Kadavath-style semantic self-evaluation baseline achieving higher AUROC than shallow token statistics under direction-agnostic scoring.

Our primary contribution is an empirical evaluation framework that makes protocol-induced inflation explicit and reproducible. Our practical contribution is a set of protocol recommendations and analysis code for future work, especially for studies that report lightweight token-level signals as baselines or as components of larger verification pipelines. Figure 1 summarizes the core findings.

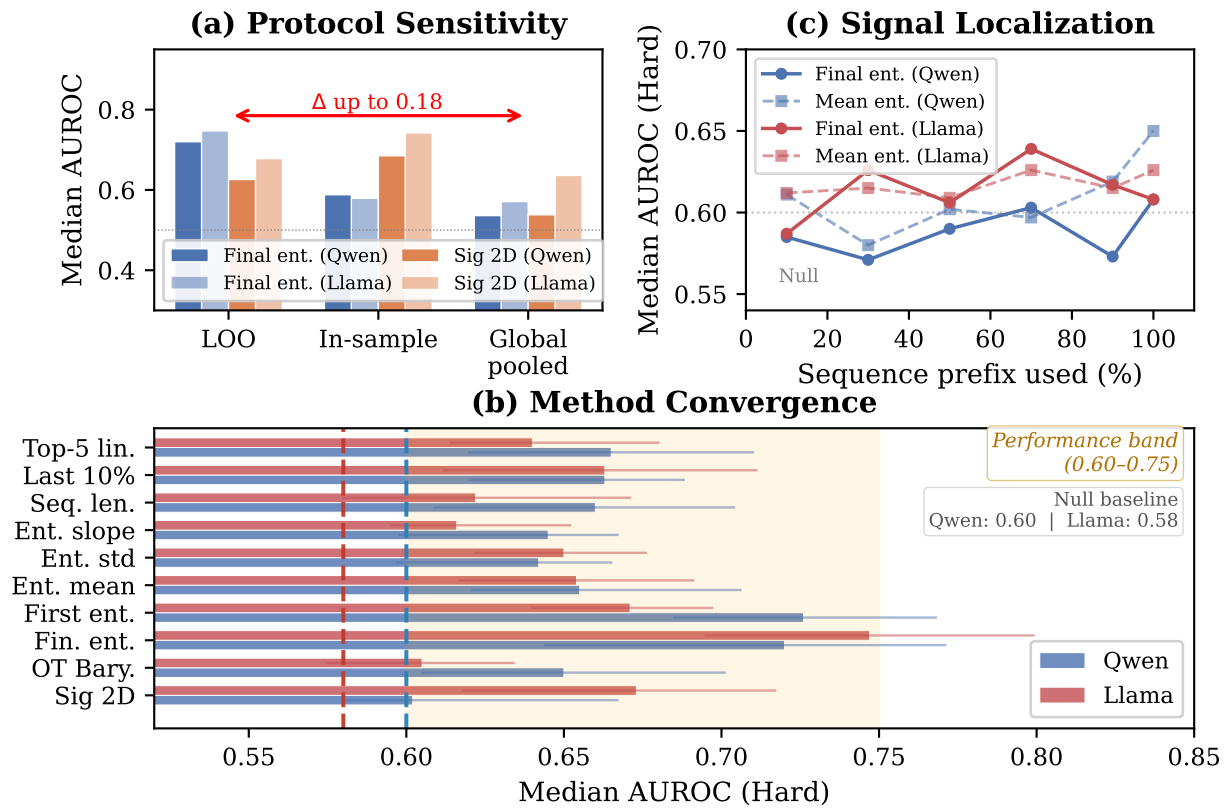

Figure 1: Overview of core findings. **(a)** Median AUROC under three evaluation protocols. **(b)** Hard-problem AUROC of all methods under LOO, with permutation-null baselines. **(c)** Prefix-based ablation showing how performance changes with sequence coverage.

2 Related Work

Outcome and process reward models. Outcome reward models (ORMs) train binary classifiers on complete solution traces Cobbe et al. [2021]. Process reward models (PRMs) extend this to step-level verification Lightman et al. [2023], Uesato et al. [2022], showing that fine-grained supervision improves verification accuracy. Both require substantial labeled data and model fine-tuning. Our work focuses on the complementary question: what can token-level statistics available from generation metadata—already available from generation—provide without any training?

Token-level uncertainty methods. Several lines of work extract uncertainty or verification-related signals from generation metadata. Wang et al. Wang et al. [2025] study entropy-after-thinking signals for reasoning-model early exit; in our experiments we adapt the resulting entropy-trajectory intuition to within-problem correctness ranking. Malinin & Gales Malinin & Gales [2021] decompose token-level uncertainty into total, data, and knowledge uncertainty (mutual information) using ensemble disagreement. Self-consistency reranking Wang et al. [2023] uses answer agreement as a proxy for correctness. A different approach is taken by Kadavath et al. Kadavath et al. [2022], who demonstrate that models can self-assess via P(True) prompts with few-shot examples. These methods are evaluated under heterogeneous protocols—including global pooling and in-sample evaluation—and generally do not report the combination of within-problem LOO, fixed-direction AUROC, and permutation-null calibration used here, making cross-study comparison

Problem: A fair six-sided die is rolled 720 times. What is the expected number of times that a prime number appears?
Solution A (correct, entropy 0.012): "The prime numbers on a six-sided die are 2, 3, and 5, so $p = 3/6 = 1/2$. Expected count: $720 \times 1/2 = \boxed{360}$."
Solution B (incorrect, entropy 0.008): "Primes: 2, 3, 5. $E[X] = np = 720 \times 3/6 = \boxed{1080}$."
Solution C (incorrect, entropy 0.341): "The probability of getting a prime is $3/6 = 1/2$. We need to count the primes in 720 rolls... The answer is $\boxed{480}$."

Figure 2: Motivating example. Solutions A (correct) and B (incorrect) both produce very low entropy, while C (incorrect) produces high entropy. The key point is that low entropy can accompany confident errors (B), so a single global direction for entropy-based verification is unreliable.

unreliable.

Internal vs. external signals. Chen et al. Chen et al. [2025] study neuron-agreement signals as internal evidence of whether LLM answers are correct. This line of work raises a complementary question: what do external token statistics retain when internal activations are unavailable? We quantify this narrower external-signal setting under controlled evaluation.

Evaluation methodology. Our contribution is to quantify these evaluation-protocol choices for shallow token-level math-reasoning verification. We use path signatures Lyons [1998], Chen [1958], Kidger et al. [2019], Chevyrev & Kormilitzin [2016] and optimal transport Peyré & Cuturi [2019] as diagnostic probes to map the performance envelope.

3 A Motivating Example

Figure 2 illustrates the fundamental challenge. On the same math problem, a correct solution and an incorrect solution can both produce very low final-token entropy when the model is confident in its (possibly wrong) answer. Conversely, an incorrect solution can produce high entropy when the model is genuinely confused. A token-level verifier faces an inherently noisy signal: the entropy-correctness relationship varies across problems, and no single global direction ("higher entropy = more likely wrong") holds universally.

We therefore evaluate the discriminative capacity of shallow token statistics after removing these artifacts. This requires an evaluation protocol that does not itself inflate or distort the observed signal.

4 Standard Evaluation Protocol

For clarity and reproducibility, we use the following six evaluation controls:

Step 1: Data Collection & WP-Eligibility Filter. For each problem, collect N independent generations (we use N=64, nucleus sampling p=0.95, temperature 0.6). Filter to within-problem eligible (WP-eligible) problems: those with ≥ 2 correct and ≥ 2 incorrect runs. This filtering is necessary for LOO evaluation and introduces a conditional subset bias.

Step 2: Within-Problem Leave-One-Out (LOO). For each problem, hold out one run as test; construct scoring centroids from the remaining correct and incorrect runs. This eliminates

problem-identity leakage.

Step 3: GroupKFold by Problem. For cross-problem analyses, split by problem ID (not runs) using GroupKFold (K=5), preventing the model from learning problem-specific difficulty.

Step 4: Direction-Agnostic Scoring with Permutation-Null Calibration. Report per-problem $\max(\text{AUROC}, 1-\text{AUROC})$, which is an *oracle upper bound* on separability. Compute permutation null via 1000 label permutations per problem; the null is ≈0.58–0.60 (not 0.5). Report both DA and fixed-direction results (Step 6).

Step 5: Per-Difficulty Stratification with Bootstrap CI. Report results stratified by difficulty tier. Compute median and 95% CI via 10,000 problem-level bootstrap resamples.

Step 6: Fixed-Direction Baseline. Learn a global scoring direction from GroupKFold training folds; evaluate on held-out problems. This provides a non-oracle estimate of practical verifier performance. Report alongside DA results.

5 Experimental Setup

5.1 Datasets and Models

Table 1: Datasets and models. WP-elig. = within-problem eligible (≥ 2 correct, ≥ 2 wrong).

Dataset	Model	Arch.	Traces	Accuracy	WP-elig. (H/M/E)
MATH	Qwen3-32B	Qwen	∼11.7K	7–98%	365 total
BigMath	Qwen2.5-Coder-7B	Qwen	25,600	55.3%	62/70/47
BigMath	Llama-3.1-8B	Llama	25,600	39.3%	64/101/83

Three model instances across two architecture families (Table 1). BigMath is our curated subset of 400 problems from the Open-R1 dataset Open-R1 [2025], including MATH Hendrycks et al. [2021], Orca Math Mukherjee et al. [2023], and GSM8K Cobbe et al. [2021]. Difficulty tiers are stratified by Llama-8B solve rate. We collect 64 independent generations per problem; correctness is determined by exact match of the extracted final answer.

5.2 Methods

Scalar baselines: entropy mean, std, final-token entropy, first-token entropy, slope, sequence length, last-10%-mean.

Temporal methods: Path signatures (depth-3, 2D $[t, \text{entropy}]$, 15 dimensions, LOO centroid-distance scoring).

Distributional methods: OT Barycenter (1D Wasserstein, quantile space, LOO scoring).

Spectral/transition methods: Fourier spectral (4D) and Markov 4-state transition matrix (16D), described and evaluated in Appendix A.2.

Cross-problem learned verifier: 3-layer MLP (6 features → 64 → 32 → 1), GroupKFold 5-fold.

6 Results

6.1 Evaluation Protocol Artifacts Account for the Largest Share of Reported Variation

Table 2: Effect of evaluation protocol on hard-problem AUROC. Negative Δ indicates degradation relative to LOO.

Method	Protocol	Qwen Hard	Llama Hard	Δ from LOO
Final entropy	LOO	0.720	0.747	—
	In-sample	0.588	0.579	−0.132 to −0.168
	Global pooled	0.536	0.571	−0.175 to −0.184
Signature 2D	LOO	0.626	0.678	—
	In-sample	0.685	0.742	+0.060 to +0.064
	Global pooled	0.538	0.636	−0.042 to −0.087

In our controlled setting, evaluation protocol choices account for a larger share of observed AUROC variation than method choice (Table 2). Pooled evaluation shifts AUROC by 0.04–0.18 relative to LOO; a plausible explanation is that pooled evaluation allows the scorer to benefit from problem-level difficulty signals rather than isolating per-solution discrimination. In-sample evaluation shifts AUROC by −0.13 to −0.17 for entropy scores but by +0.06 for signatures. The entire range of method-driven variation under LOO (0.60–0.75, span 0.15) is smaller than the protocol-induced shift from pooled evaluation (up to 0.18).

6.2 Under Controlled Evaluation, Shallow Methods Fall into an Overlapping Range

Table 3 summarizes the LOO results. Under controlled LOO evaluation, the analyzed methods produce DA AUROC 0.60–0.75 with 95% CIs that substantially overlap. The genuine excess above the permutation null is only ≈0.05–0.15. Note that absence of statistical significance does not establish equivalence; rather, our sample size limits the minimum detectable effect to ≈0.12 AUROC (see Section 7.2).

Oracle vs. practical verification. Direction-agnostic (DA) scoring is a per-problem oracle: it selects the better scoring direction post-hoc. A fixed-direction baseline, learning one global direction from GroupKFold training folds, reveals that DA scoring inflates scalar-method AUROC by 0.02–0.27. For final-token entropy, fixed-direction AUROC drops to 0.47–0.48 (below chance), because the entropy-correctness direction varies idiosyncratically across problems with no learnable global pattern (direction accuracy: 41–47%). Distance-based methods (Signature, OT) are more robust, with gaps of only 0.03–0.07 and direction accuracy 66–73%. This indicates that high DA AUROC for positional scalars reflects per-problem direction selection, and DA scoring should be interpreted as an upper bound.

Table 3: LOO results on hard problems (DA and fixed-direction AUROC) with 95% bootstrap CIs.

Method	Qwen Hard (n=62)		Llama Hard (n=64)	
	DA Median	Fixed-Dir	DA Median	Fixed-Dir
Signature 2D	0.602 [0.581,0.667]	0.595	0.673 [0.618,0.717]	0.597
OT Barycenter	0.650 [0.605,0.701]	0.577	0.605 [0.575,0.634]	0.576
Final entropy	0.720 [0.644,0.771]	**0.470**	**0.747** [0.695,0.799]	**0.481**
First entropy	**0.726** [0.685,0.768]	0.500	0.671 [0.640,0.697]	0.500
Entropy mean	0.655 [0.621,0.706]	0.533	0.654 [0.617,0.691]	0.479
Entropy std	0.642 [0.597,0.665]	—	0.650 [0.622,0.676]	—
Entropy slope	0.645 [0.598,0.667]	—	0.616 [0.595,0.652]	—
Sequence length	0.660 [0.609,0.704]	—	0.622 [0.580,0.671]	—
Last-10%-mean	0.663 [0.620,0.688]	—	0.663 [0.612,0.711]	—
Top-5 linear	0.665 [0.620,0.710]	—	0.640 [0.614,0.680]	—
Permutation null	≈0.60		≈0.58	

6.3 Reproducing Prior Token-Level Verification Methods

To compare our findings against related published methods, we implement three prior-inspired baselines under our SEP protocol on Qwen2.5-Coder-7B Hard (n=62). These are controlled adaptations rather than exact replications of the original experimental settings.

Table 4: Reproduced prior methods under SEP (Qwen Hard, n=62).

Method	DA Median AUROC	95% CI	Flip Rate
Wang et al. (2025): adapted entropy-trajectory baseline			
Violation count ($-v$)	0.722	[0.700, 0.743]	95.2%
Entropy range ($H_0 - H_N$)	0.652	[0.626, 0.688]	80.6%
Malinin & Gales (2021): Token-level MI/RMI			
LOO centroid distance	0.670	[0.619, 0.705]	12.9%
LOO KL divergence	0.651	[0.617, 0.703]	66.1%
Mean entropy (MC)	0.647	[0.621, 0.674]	62.9%
Kadavath et al. (2022): P(True) self-evaluation			
Zero-shot P(True)	**0.911**	[0.857, 0.964]	97.3%
Few-shot P(True)	**0.865**	[0.764, 0.952]	27.9%
Reference: our methods (same model/tier)			
Final entropy (DA oracle)	0.720	[0.644, 0.771]	75.8%
Signature 2D	0.602	[0.581, 0.667]	46.8%

Wang et al. Wang et al. [2025]: Inspired by entropy-after-thinking style signals, we split each trace into 4 checkpoint positions (25/50/75/100% of paragraphs), sample m=5 continuations per checkpoint, extract answer distribution entropy, and compute monotonicity violation count. The violation count achieves DA 0.722, within the band, with 95.2% flip rate (the highest of any method).

Malinin & Gales Malinin & Gales [2021]: Using the top-20 log-probabilities already stored in

our NPZ files, we compute LOO ensemble posterior divergence (KL, centroid distance) across 64 runs. All variants fall at 0.647–0.670, within the band. Ensemble-level MI and RMI are identical for all runs within a problem (they measure a property of the problem, not individual traces), and thus have no per-run discriminative power (AUROC = 0.500).

Kadavath et al. Kadavath et al. [2022]: Unlike the shallow entropy features above, P(True) prompts the model to evaluate its own reasoning for logical coherence, accessing semantic content rather than just statistical properties. Accordingly, zero-shot P(True) achieves 0.911 DA AUROC, well above the band. Its 97.3% flip rate should be interpreted cautiously: it shows that the raw orientation of this score is unstable under our per-problem DA convention, not that P(True) provides a reliable fixed-direction deployment signal. Despite this orientation instability, the magnitude of the P(True) separability is sufficiently large that DA AUROC remains high. This result suggests a distinction, in this setting, between shallow statistical features and semantic self-evaluation under the same oracle-style DA metric.

6.4 Cross-Problem Learning Yields Modest Gains

Table 5: Cross-problem GroupKFold (5-fold) MLP results by difficulty tier.

Model	Easy	Medium	**Hard**	All
Llama-3.1-8B	0.815 [0.782,0.844]	0.740 [0.692,0.768]	**0.726** [0.683,0.748]	0.750
Qwen2.5-Coder-7B	0.810 [0.724,0.863]	0.689 [0.649,0.738]	**0.660** [0.601,0.712]	0.702

Cross-problem learning remains within the same AUROC range on hard problems (Table 5). Most of the MLP's benefit comes from easier problems, indicating that the hard-problem band reflects genuine information limitations.

6.5 Signal is Graded Across the Sequence

Discriminative signal is graded across the token sequence, not sharply concentrated at any single position. A controlled ablation shows: (1) the post-answer region (after `\boxed{`) produces the strongest single-position signal (correct solutions: lower post-answer entropy); (2) the global entropy level (captured by the mean) is also informative, with 10% of the sequence already achieving 0.59–0.61 AUROC (Table 6); (3) masking the last 5 tokens slightly *improves* AUROC (final formatting tokens are noisy); (4) a separate boxed-answer ablation confirms that pre-answer final-entropy matches full-sequence AUROC within 0.02, ruling out formatting artifacts as the primary signal source.

Note that Table 6 reports in-sample (non-LOO) AUROC for the prefix truncation experiment, which is a different evaluation from the LOO-based results in Table 3; the two tables are not directly comparable. The "within 0.02" claim refers to the boxed-answer ablation (pre-answer vs. full-sequence entropy under the same evaluation), not to Table 6.

In summary, the answer-finalization region carries the strongest localized signal, but the overall entropy level throughout the generated solution provides a broad, stable baseline. These two signals are partially redundant, which is why the entropy mean from just 10% of the sequence approaches full-sequence performance.

Table 6: Prefix truncation on hard problems; in-sample AUROC using first $k/10$ of the sequence.

% used	Qwen Hard		Llama Hard	
	Final-ent	Mean-ent	Final-ent	Mean-ent
10%	0.585	0.611	0.587	0.612
50%	0.590	0.602	0.606	0.609
100%	0.608	0.650	0.608	0.626

6.6 Practical Context: Self-Consistency and Pass@k

Table 7: Majority voting accuracy and pass@k on hard problems.

Model	Tier	Maj. Acc.	pass@4	pass@16	pass@64
Qwen	Hard (140)	27.1%	39.6%	47.0%	52.9%
Llama	Hard (140)	3.6%	24.4%	42.4%	55.0%

On hard problems, pass@64 reaches 53–55% (Table 7), meaning a perfect verifier could find correct solutions for half the problems. A verifier with 0.72 DA AUROC can meaningfully improve best-of-N selection but cannot reliably identify the single best solution.

7 Discussion

What the band means. The DA AUROC range of 0.60–0.75 characterizes the practical performance envelope of shallow token features under within-problem evaluation. This result is robust across model families, but the band should not be interpreted as an information-theoretic ceiling: it is specific to shallow statistical features on WP-eligible hard problems, modulated by model capability, and partially metric-induced (DA null ≈0.58–0.60).

Oracle vs. practical performance. The most important nuance for practitioners is the gap between DA (oracle) and fixed-direction AUROC. Final-token entropy has the highest DA AUROC (0.72–0.75) but the *lowest* fixed-direction AUROC (0.47–0.48). This means its apparent superiority under DA evaluation is an artifact of per-problem direction selection, not a stable verification signal. Distance-based methods (Signature, OT) provide more reliable practical performance (fixed-direction gaps of only 0.03–0.07).

When token features work. Token features are most informative in our experiments when model accuracy is high (Easy: AUROC around 0.83). They are insufficient as standalone verifiers when models operate near their capability frontier on hard problems, but may still be useful as low-cost triage features, calibration diagnostics, or ablation baselines inside stronger verification systems. Semantic evaluation, as in the P(True) baseline, should be treated as a richer signal source rather than a shallow token-statistic feature.

Relation to stronger verification pipelines. Our results should not be read as arguing against verification for AI-for-math systems. Rather, they clarify where shallow token statistics sit in the design space. Symbolic checking, tool execution, learned verifiers, reward models, process supervision, and semantic self-evaluation can access information unavailable to external entropy

traces. The role of SEP is to make comparisons with these richer methods fair: a method that beats entropy under global pooling or direction-agnostic scoring may not beat a properly controlled token-level baseline.

7.1 Methodological Recommendations

We recommend the SEP (Section 4) as a reporting checklist for comparable evaluations:

1. **Adopt within-problem LOO evaluation.**
2. **Use GroupKFold by problem for cross-problem analyses.**
3. **Report both DA and fixed-direction AUROC.**
4. **Report per-difficulty breakdowns with bootstrap CIs.**
5. **Report permutation-null baselines.**
6. **Compare against final-token entropy under LOO.**

7.2 Limitations & Open Questions

Scope. We evaluate three models at 7B–32B scale on mathematical word-problem benchmarks. With $n \approx$62–64 hard problems per model, at 80% power and α=0.05, the minimum detectable effect is $\approx$0.12 AUROC. Reasoning-specialized architectures, multimodal math settings, and structured generations outside text-only math remain untested. The SEP controls are not specific to word problems, but whether the same empirical band appears in those settings is an open question.

WP-eligible selection bias. WP-eligibility excludes problems where a model is always correct or always wrong. On hard problems, this excludes $\approx$56% of problems (64 of 140 for Llama). The excluded problems are disproportionately those where the model has very low accuracy; results apply specifically to problems where within-problem evaluation is feasible.

Formatting artifacts. A separate boxed-answer ablation confirms the entropy signal is not driven by answer-formatting artifacts: restricting entropy computation to pre-answer tokens (before `\boxed{`) yields DA AUROC within 0.02 of the full-sequence baseline under the same evaluation protocol.

Open questions. Does the band persist for reasoning-specialized models? Can sequence-based architectures with cross-problem training substantially improve on MLP results? Would larger ensembles narrow CIs to reveal method differences? Can the same controls be applied to multimodal or structured mathematical outputs where correctness depends on diagrams, code execution, or formal proof states?

8 Conclusion

This study provides a controlled baseline for evaluating token-level verification of LLM math reasoning. Under controlled evaluation, shallow external token statistics occupy a practical performance envelope of DA AUROC 0.60–0.75 on WP-eligible hard problems. The gap between oracle and

fixed-direction performance is substantial for positional scalars, indicating that high apparent separability often reflects per-problem direction selection rather than a stable verification signal. Within the scope of our study, increasing feature complexity alone does not appear sufficient to substantially improve standalone verification on hard math problems. This motivates exploring richer signal sources, such as semantic self-evaluation, internal representations, tool execution, or process supervision. The accompanying repository provides the evaluation scripts and paper artifacts.

Reproducibility

Analysis code and evaluation scripts are released with this arXiv version; large generated traces are not included in the source package. Bootstrap CIs: 10,000 resamples; permutation null: 1,000 iterations per problem. Prior-inspired baselines use the same Qwen2.5-Coder-7B model and BigMath dataset.

A Additional Methods and Results

A.1 Direction-Flip Rate Analysis

Table 8: Direction-flip rate and DA AUROC on hard BigMath problems (64 runs).

Method	Flip Rate		DA Median	
	Qwen	Llama	Qwen	Llama
Signature 2D	46.8%	29.7%	0.602	0.673
OT Barycenter	38.7%	56.2%	0.650	0.605
Fourier spectral	0.0%	0.0%	0.627	0.668
Markov 4×4	0.0%	0.0%	0.680	0.648
Final entropy	75.8%	93.8%	0.720	0.747
First entropy	74.2%	95.3%	0.726	0.671
Entropy mean	61.3%	60.9%	0.655	0.654
Entropy std	64.5%	67.2%	0.642	0.650
Entropy slope	64.5%	70.3%	0.645	0.616
Length	62.9%	50.0%	0.660	0.622
Top-5 linear	38.7%	28.1%	0.665	0.640

A.2 Spectral and Transition Methods

To probe whether richer temporal representations improve verification, we evaluate two additional methods: (1) **Fourier spectral features** (4D), computed as the magnitudes of the first 4 discrete Fourier transform coefficients of the entropy sequence, capturing periodic patterns in entropy oscillations; and (2) **Markov transition matrices** (16D), computed by discretizing entropy into 4 quantile-based states and counting state transitions, capturing the dynamics of entropy regime changes. Both fall within the same band as all other methods (Table 9), confirming that increasing temporal representational complexity does not qualitatively improve verification on hard problems.

Table 9: Spectral and transition method results on hard BigMath (LOO, direction-agnostic).

Method	Qwen Hard (n=62)		Llama Hard (n=64)	
	Median	95% CI	Median	95% CI
Fourier spectral (4D)	0.627	[0.596,0.692]	0.668	[0.614,0.696]
Markov 4×4 (16D)	0.680	[0.650,0.710]	0.648	[0.616,0.707]
Reference				
Signature 2D (15D)	0.602	[0.581,0.667]	0.673	[0.618,0.717]
Final entropy (1D)	0.720	[0.644,0.771]	0.747	[0.695,0.799]

A.3 Pairwise Significance Tests

No pairwise comparison between geometric and scalar methods reaches significance on Llama Hard (n=64): all yield $p > 0.10$ (paired Wilcoxon). Most pairwise differences fall within ±0.05 AUROC, suggesting limited practical separation given the study's statistical power.

A.4 Prior Method Reproduction Details

Wang et al. Wang et al. [2025]: For our adapted entropy-trajectory baseline, each trace is split into checkpoint positions at 25/50/75/100% of paragraph boundaries. At each checkpoint, we sample m=5 continuations (temperature 0.7, max 150 tokens) using the Qwen2.5-Coder-7B model. Answers are extracted via `\boxed{}` pattern matching. Entropy trajectory monotonicity is evaluated with ε=0.01 tolerance. Total: 78,320 generation calls.

Malinin & Gales Malinin & Gales [2021]: Using top-20 log-probabilities from NPZ files (already collected during generation), we compute LOO ensemble posterior distributions across 64 runs per problem. No additional model calls needed.

Kadavath et al. Kadavath et al. [2022]: Zero-shot P(True) prompts present the model's own reasoning trace and proposed answer, asking for True/False judgment. Few-shot adds 3 brainstorming answers from other runs of the same problem. Greedy decoding (temperature 0, max 10 tokens). Total: 3,968 zero-shot + 3,968 few-shot calls.

References

Chen, K.-T. (1958). Integration of paths. *Trans. AMS*, 89(2), 395–407.

Chen, Y. et al. (2025). Do LLMs Signal When They're Right? Evidence from Neuron Agreement. *arXiv:2510.26277.*

Chevyrev, I. & Kormilitzin, A. (2016). A primer on the signature method. *arXiv:1603.03788.*

Cobbe, K. et al. (2021). Training verifiers to solve math word problems. *arXiv:2110.14168.*

Hendrycks, D. et al. (2021). Measuring mathematical problem solving. *NeurIPS 2021.*

Kadavath, S. et al. (2022). Language models (mostly) know what they know. *arXiv:2207.05221.*

Kidger, P. et al. (2019). Deep signature transforms. *NeurIPS 2019*.

Lightman, H. et al. (2023). Let's verify step by step. *ICLR 2024*.

Lyons, T. (1998). Differential equations driven by rough signals. *Rev. Mat. Iberoamericana*, 14(2), 215–310.

Malinin, A. & Gales, M. (2021). Uncertainty estimation in autoregressive structured prediction. *ICLR 2021*.

Mukherjee, S. et al. (2023). Orca: Progressive learning from complex explanation traces. *arXiv:2306.02707*.

Open-R1. Big-Math-RL-Verified-Processed. https://huggingface.co/datasets/open-r1/Big-Math-RL-Verified-Processed, 2025.

Peyré, G. & Cuturi, M. (2019). Computational optimal transport. *FnT in ML*, 11(5-6), 355–607.

Uesato, J. et al. (2022). Solving math word problems with process- and outcome-based feedback. *arXiv:2211.14275*.

Wang, X. et al. (2023). Self-consistency improves chain of thought reasoning in language models. *ICLR 2023*.

Wang, X. et al. (2025). Entropy After `</think>`: Early Exit Through Confidence of Reasoning in Large Reasoning Models. *arXiv:2509.26522*.