Beyond a Single Temperature: Comparable Trace Scoring for Multi-Temperature Test-Time Reasoning

Anonymous ACL submission

Abstract

Test-time scaling improves LLM reasoning by generating multiple candidate traces and aggregating their answers. Sampling at several temperatures can broaden this pool, but probability-based scores obtained at different temperatures are not directly comparable. We study this cross-temperature scoring problem and introduce Temperature-Normalized Mean Absolute Deviation (TN-MAD), a top-k log-probability dispersion score whose direct temperature factor cancels in local comparisons. We use TN-MAD to filter a mixed-temperature pool and weight the final answer vote. When comparing against baselines including majority voting, DeepConf, and TURN, TN-MAD achieves 90.0% accuracy on BRUMO25 (+7.3 percentage points over the best baseline) and gains 3.3 points on HMMT25 with Qwen3-4B-Thinking-2507. Its applicability is further demonstrated by its ability to act as a plug-in component within TURN-anchored and DeepConf-style pipelines. These results demonstrate that temperature-comparable trace scoring can effectively exploit complementary reasoning traces across sampling temperatures. Our codes are available at https://anonymous.4open.science/r/anonymous-TN-MAD-B430/.

1 Introduction

Test-time scaling (TTS) improves LLM reasoning through additional computation during inference (Brown et al., 2024; Wang et al., 2023). It has two coupled parts: *exploration* generates a pool of reasoning trajectories, and *aggregation* selects or combines their answers. Existing pipelines usually implement aggregation with majority voting, Best-of-N selection, or confidence-aware ranking and filtering (Wang et al., 2023; Cobbe et al., 2021; Fu et al., 2026b).

The focus on aggregation leaves an upstream issue unresolved: a selector cannot choose a useful trajectory that was not generated. Self-consistency aggregates answer frequencies, Best-of-N ranks completed trajectories, and confidence-aware methods filter generated traces (Wang et al., 2023; Cobbe et al., 2021; Fu et al., 2026b). Each method works on a candidate pool generated by a commonly fixed configuration. The coverage of that pool can therefore limit test-time reasoning, independently of the performance of aggregation rules.

As a critical sampling parameter, temperature significantly influences candidate coverage. Lower temperatures favor conservative, high-probability paths, whereas higher temperatures promote the exploration of diverse, alternative solutions (Wu and Tambe, 2025; Zhang et al., 2024; Zhu et al., 2024). Relying on a fixed temperature may miss useful traces, since the oracle best temperature $T^*(x)$ is highly problem-specific, which is corroborated by the results in Figure 1. While sampling from different temperatures could broaden coverage, it complicates scoring. Existing metrics such as log probability, entropy, and top-k confidence are computed from temperature-dependent distributions (Cobbe et al., 2021; Fu et al., 2026b), resulting in a preference on low-temperature results. Such systemic bias makes existing methods incompatible with cross-temperature sampling, failing to recognize instances where high-temperature samples hold a distinct advantage as shown in Figure 1.

To formulate and overcome the aforementioned limitations, we first theoretically explain the inherent bias in log-probability-based metrics. We then propose Temperature-Normalized Mean Absolute Deviation (TN-MAD) as a robust traces scoring solution for mixed-temperature pools. TN-MAD measures top-k log-probability dispersion, with the direct source-temperature term removed from local comparisons. This normalization enables reasoning traces generated at different temperatures to be fairly ranked under a unified metric. We integrate TN-MAD into a comprehen-

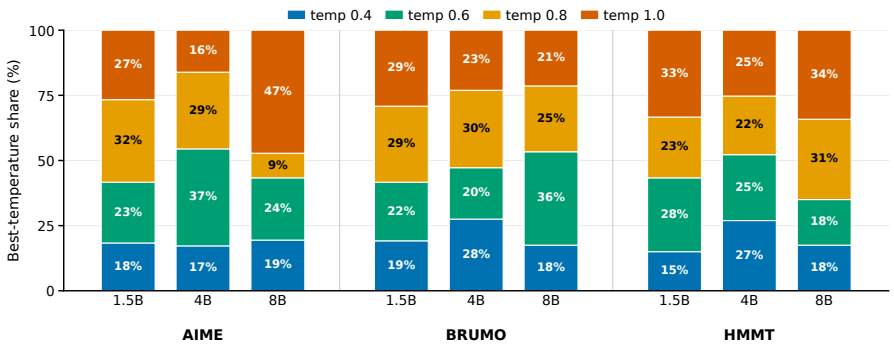


Figure 1: Problem-level temperature heterogeneity across datasets and models. The oracle best Pass@1 temperature varies substantially across settings, motivating mixed-temperature candidate pools rather than a single fixed sampling temperature. Ties are split evenly among tied temperatures.

sive inference pipeline featuring trace filtering and scaled majority voting. Evaluated across AIME25, HMMT25, and BRUMO25 with three reasoning models, our method demonstrates substantial improvements. Notably, using the Qwen3-4B-Thinking-2507 model, the pipeline attains 90.0% accuracy on BRUMO25 (+7.3 percentage points over the best baseline) and gains 3.3 points on HMMT25. Furthermore, TN-MAD exhibits strong generalizability, serving effectively as a plug-in component for TURN-anchored temperature selection and confidence-filtering pipelines.

The main contributions are:

1. **Problem formulation.** We identify two coupled limitations of current TTS pipelines: fixed-temperature sampling restricts the coverage of trajectories, while existing probability-based scores fail to rank trajectories generated across different temperatures on a comparable scale.
2. **Theoretical analysis and method.** We theoretically characterize the temperature dependence of widely used scoring rules (e.g. log probability, entropy). To address this, we propose TN-MAD, a top-k log-probability dispersion score that provably eliminates the direct temperature factor under fixed candidates and full-softmax log probabilities, yielding a temperature-invariant signal for trace ranking.
3. **Empirical validation.** We instantiate TN-MAD in a multi-temperature inference pipeline and evaluate it across three competition-level reasoning benchmarks (AIME25, HMMT25, BRUMO25) with three reasoning models. TN-MAD outperforms fixed-temperature aggregation baselines under matched answer-aggregation budgets.

2 Related Work

Test-time scaling for reasoning. Increasing inference-time computation can improve reasoning performance. Chain-of-thought (Wei et al., 2022) and self-consistency (Wang et al., 2023) provide foundations for sampling reasoning traces and aggregating answers through majority voting. Best-of-N instead selects candidates using external verifiers or reward models (Cobbe et al., 2021). As correct traces become more likely to appear with broader sampling (Brown et al., 2024; Welleck et al., 2024), the central problem is to explore sufficiently diverse candidates while selecting reliable ones. One line of work ranks or filters candidates with probability-based signals from the model itself, including token confidence, log probability, and entropy (Kang et al., 2025; Geng et al., 2024). DeepConf (Fu et al., 2026b) and Certaindex (Fu et al., 2026a) use certainty estimates for efficient trace filtering and early stopping. However, these signals are often poorly calibrated and can conflict with the diversity needed for broad exploration, since high-confidence generations tend to be conservative (Li et al., 2026; Geng et al., 2024).

Temperature-aware test-time inference. Temperature controls distribution sharpness and thereby affects both generation quality and diversity (Xie, 2017; Fan et al., 2018; Holtzman et al., 2020; Minh et al., 2025). In reasoning tasks, higher temperatures can reveal candidate paths missed by conservative decoding, while the specific optimal temperature setting may differ across instances (Wu and Tambe, 2025; Salah and Muneer, 2026). Several methods therefore select or calibrate temperature during test-time inference. TURN estimates near-optimal temperatures without validation labels (Du

et al., 2025); CarBoN provides input-specific calibration for Best-of-N (Tang et al., 2025); and other methods adjust temperature at the decoding-step or search-tree level (Zhang et al., 2024; Troshin et al., 2025; Cai et al., 2024). These approaches primarily regulate candidate generation. A candidate pool drawn from multiple temperatures raises a different problem: its scoring rule must remain strictly comparable across temperature settings.

3 Preliminary

3.1 Temperature-Controlled Sampling

Temperature controls the sharpness of the decoding distribution in neural text generation (Xie, 2017). At decoding step t, given model logits $z_t \in \mathbb{R}^{|\mathcal{V}|}$, the temperature-scaled sampling distribution is

$$p_T(v) = \frac{\exp(z_t(v)/T)}{\sum_{u \in \mathcal{V}} \exp(z_t(u)/T)}. \quad (1)$$

Lower temperatures concentrate probability mass on high-logit tokens, whereas higher temperatures flatten the distribution and increase sampling diversity (Xie, 2017; Fan et al., 2018; Holtzman et al., 2020; Minh et al., 2025).

For an input x sampled at a fixed temperature T, let

$$\mathcal{Y}_T(x) = \{y_1, y_2, \ldots, y_n\}, \quad y_i \sim p_T(\cdot \mid x)$$

be the sampled candidate set.

3.2 Problem-Level Temperature Heterogeneity

Prior work reports that different sampling temperatures can solve different subsets of reasoning problems (Wu and Tambe, 2025). We capture this problem-level variation with an oracle defined over a fixed candidate set of temperatures:

$$T^*(x) = \arg\max_{T \in \mathcal{T}} \widehat{\mathrm{Acc}}(x, T),$$

where $\mathcal{T}$ denotes the candidate temperature set, and $\widehat{\mathrm{Acc}}(x, T)$ denotes the empirical Pass@1 success rate for problem x under temperature T, estimated from sampled traces generated at that temperature. In our analysis, $\mathcal{T} = \{0.4, 0.6, 0.8, 1.0\}$.

Figure 1 confirms this problem-level heterogeneity: $T^*(x)$ varies across problems, and no single temperature dominates across datasets and model sizes, consistent with prior findings on temperature diversity in reasoning (Wu and Tambe, 2025).

This variation motivates sampling from multiple temperatures rather than committing to one global setting. We therefore form the mixed candidate pool

$$\mathcal{Y}_{\mathcal{T}}(x) = \bigcup_{T \in \mathcal{T}} \mathcal{Y}_T(x). \quad (2)$$

This construction broadens candidate coverage without requiring access to the unknown oracle temperature $T^*(x)$.

3.3 Basic Best-of-N Scoring Methods

Multi-sample inference generates candidate traces and then selects or aggregates a final answer (Brown et al., 2024; Welleck et al., 2024). In Best-of-N selection, a scoring criterion S determines the selected trace (Cobbe et al., 2021; Brown et al., 2024):

$$\hat{y} = \arg\max_{y \in \mathcal{Y}_T(x)} S(y, x).$$

Many test-time scaling methods derive such scores from model-internal probability signals (Fu et al., 2026b; Kang et al., 2025; Geng et al., 2024). Let $\{p_{t,1}(T), p_{t,2}(T), \ldots, p_{t,k}(T)\}$ denote the top-k probabilities at step t under temperature T. We consider two token-level quantities:

Top-k Entropy (Fu et al., 2026b; Zhang et al., 2024; Basu et al., 2021).

$$H_t(T) = -\sum_{j=1}^{k} p_{t,j}(T) \log p_{t,j}(T).$$

Top-k Confidence (Fu et al., 2026b).

$$C_t(T) = -\frac{1}{k} \sum_{j=1}^{k} \log p_{t,j}(T).$$

High entropy indicates greater local uncertainty, whereas higher confidence indicates greater local model certainty. In confidence-based filtering, the corresponding confidence direction is used for ranking (Fu et al., 2026b; Zhang et al., 2024; Basu et al., 2021). Aggregating these token-level quantities yields trace-level signals for Best-of-N selection or confidence filtering (Fu et al., 2026b; Kang et al., 2025; Fu et al., 2026a).

3.4 Temperature-Dependent Scores in Cross-Temperature Best-of-N

Cross-temperature sampling changes the scoring problem because candidates are evaluated under different temperature-scaled distributions. For example, the top-k negative log-confidence cost is

$$C_t(T) = \log Z_t(T) - \frac{1}{kT}\sum_{j=1}^{k} z_{t,j},$$

where $Z_t(T)$ is the temperature-scaled partition function at step t,

$$Z_t(T) = \sum_{v\in\mathcal{V}} \exp(z_t(v)/T).$$

The value of $C_t(T)$ therefore depends directly on T, and entropy likewise changes with the temperature-scaled distribution p_T. Consequently, ranking a mixed-temperature pool with these raw scores may reflect temperature-induced scale differences rather than trace quality. Equations 8 and 9 in Appendix B provide detailed analysis. Cross-temperature inference therefore requires a scoring rule whose comparisons are not driven by the source temperature itself. We introduce such a rule in the next section.

4 Method

Let $\mathcal{Y}_{\mathcal{T}}(x)$ be a candidate pool for input x, with traces sampled at source temperatures $T \in \mathcal{T}$. The method assigns each trace a score that can be compared across these temperatures and uses that score for filtering and answer voting. We first define the token-level Temperature-Normalized Mean Absolute Deviation (TN-MAD) score and its temperature-cancellation property, then specify trace-level aggregation and voting.

Figure 2 illustrates the pipeline. For a trace $y \in \mathcal{Y}_{\mathcal{T}}(x)$ with length N and extracted answer $A(y)$, the method computes a trace score $S(y)$, retains high-scoring traces, and weights their answers in the final vote.

4.1 Temperature-Comparable Token-Level Score

At decoding step t, denote the top-k probabilities under source temperature T by $\{p_{t,1}(T), p_{t,2}(T), \ldots, p_{t,k}(T)\}$. Their mean log probability is defined as

$$\overline{\log p_t(T)} = \frac{1}{k}\sum_{j=1}^{k} \log p_{t,j}(T).$$

The token-level **Temperature-Normalized Mean Absolute Deviation (TN-MAD)** score is defined as

$$M_t(T) = \frac{T}{k}\sum_{j=1}^{k} \left|\log p_{t,j}(T) - \overline{\log p_t(T)}\right|. \quad (3)$$

This score is the mean absolute deviation of the top-k log probabilities, scaled by the source temperature. With a fixed returned top-k set and full-softmax log probabilities, a larger value means that the leading token candidates are more separated; a smaller value means that the local distribution is flatter.

The multiplication by T cancels the direct effect of temperature rescaling. For a fixed decoding step and top-k candidate set, Equation 1 gives

$$\log p_{t,j}(T) = \frac{z_{t,j}}{T} - \log Z_t(T),$$

and therefore

$$\overline{\log p_t(T)} = \frac{\bar{z}_t}{T} - \log Z_t(T), \quad \bar{z}_t = \frac{1}{k}\sum_{j=1}^{k} z_{t,j}.$$

Thus,

$$\log p_{t,j}(T) - \overline{\log p_t(T)} = \frac{z_{t,j} - \bar{z}_t}{T}.$$

Substituting into $M_t(T)$ gives

$$M_t(T) = \frac{1}{k}\sum_{j=1}^{k} \left|z_{t,j} - \bar{z}_t\right|. \quad (4)$$

Thus, for $T > 0$, $M_t(T)$ contains no direct temperature factor and preserves the ordering of fixed logit sets. The derivation assumes an unchanged top-k candidate set and full-softmax log probabilities. Appendix B contrasts this property with the temperature sensitivity of raw confidence and entropy scores.

4.2 Trace-Level Score Aggregation

We aggregate token scores to obtain one score for each trace. The main method uses tail aggregation, adapted from DeepConf (Fu et al., 2026b). Mean and bottom-10% aggregation are included as ablation variants.

Tail aggregation. Let m_{tail} be the tail length. The trace score is the mean TN-MAD score over the final part of the trace:

$$S_{\text{tail}}(y) = \frac{1}{\min(N, m_{\text{tail}})} \times \sum_{t=\max(1,N-m_{\text{tail}}+1)}^{N} M_t(T).$$

Mean aggregation. This variant averages TN-MAD over all tokens in a trace:

$$S_{\text{mean}}(y) = \frac{1}{N}\sum_{t=1}^{N} M_t(T).$$

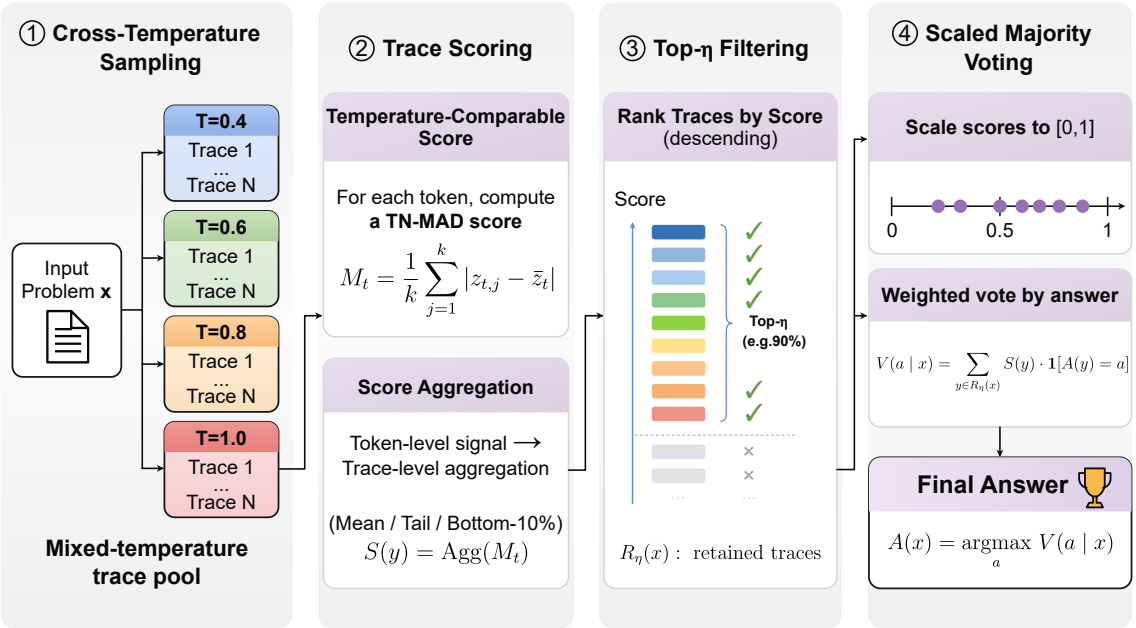


Figure 2: Overview of TN-MAD inference. Traces sampled at multiple temperatures form a mixed-temperature pool. TN-MAD scores the traces; the top-η fraction is retained and combined by scaled majority voting.

Bottom-10% group aggregation. Let $\mathcal{G}(y)$ denote the set of sliding-window token groups in trace y, with group score

$$\overline{M}(G) = \frac{1}{|G|} \sum_{t \in G} M_t(T).$$

Let $\mathcal{G}_{\text{bottom}}(y)$ contain the lowest-scoring 10% of these groups. The corresponding trace score is

$$S_{\text{bottom10}}(y) = \frac{1}{|\mathcal{G}_{\text{bottom}}(y)|} \sum_{G \in \mathcal{G}_{\text{bottom}}(y)} \overline{M}(G).$$

All principal results use tail aggregation, i.e., $S(y) = S_{\text{tail}}(y)$. Section 5.3 compares it with the mean and bottom-10% variants.

For decoding with a time-varying temperature, we apply $M_t(T_t)$ at position t. The same cancellation holds; Appendix B.3 gives the derivation.

4.3 Using TN-MAD for Filtering and Voting

Given a trace score $S(y)$ and a retention fraction $\eta \in (0, 1]$, we retain the highest-scoring η fraction of candidate traces:

$$\mathcal{R}_\eta(x) = \mathrm{Top}_\eta\left(\mathcal{Y}_\mathcal{T}(x); S\right). \tag{5}$$

Here, $\mathrm{Top}_\eta(\cdot; S)$ ranks traces by S. The default configuration sets the hyperparameter $\eta = 0.9$; Appendix C reports results for other retention fractions. Filtering removes the lowest-scoring traces before answer aggregation.

At $\eta = 0.9$, most of the pool remains after filtering. We therefore use **scaled majority voting** rather than assigning an equal vote to every retained trace. For each problem, min-max scaling maps retained scores to weights in $[0, 1]$ within $\mathcal{R}_\eta(x)$. Let

$$S_\eta^{\min}(x) = \min_{y \in \mathcal{R}_\eta(x)} S(y),$$
$$S_\eta^{\max}(x) = \max_{y \in \mathcal{R}_\eta(x)} S(y).$$

When $S_\eta^{\max}(x) > S_\eta^{\min}(x)$, a retained trace y receives weight

$$w_\eta(y) = \frac{S(y) - S_\eta^{\min}(x)}{S_\eta^{\max}(x) - S_\eta^{\min}(x)}. \tag{6}$$

If all retained traces have the same score, we set $w_\eta(y) = 1$ for every $y \in \mathcal{R}_\eta(x)$, and the rule becomes ordinary majority voting. Otherwise, traces with higher TN-MAD scores receive larger weights. The score of answer a is

$$V_{\eta,\text{scaled}}(a \mid x) = \sum_{y \in \mathcal{R}_\eta(x)} w_\eta(y) \mathbf{1}[A(y) = a]. \tag{7}$$

The final answer is selected as

$$\hat{A}_\eta(x) = \arg\max_a V_{\eta,\text{scaled}}(a \mid x).$$

5 Experiments

5.1 Experimental Setup

Models. We evaluate DeepSeek-R1-Distill-Qwen-1.5B [1] (DeepSeek-AI, 2025), Qwen3-4B-Thinking-2507 [2] (Qwen Team, 2025a), and Qwen3-8B [3] (Qwen Team, 2025b). Qwen3-8B is evaluated with thinking mode enabled.

Benchmarks. The benchmark sources are AIME25 (Art of Problem Solving, 2025a,b), HMMT25 (Harvard-MIT Mathematics Tournament, 2025), and BRUMO25 (Brown University Math Olympiad, 2025). We evaluate 30 problems from each benchmark, covering challenging mathematical reasoning problems with varied difficulty and problem styles.

Candidate budget. For accuracy comparisons, answer aggregation uses 64 traces: fixed-temperature baselines draw them all at $T = 0.6$, while our default cross-temperature setting draws 16 from each temperature in $\mathcal{T} = \{0.4, 0.6, 0.8, 1.0\}$. TURN-based methods additionally run a temperature-selection stage before answer aggregation; this overhead is reported separately in Appendix A.2.

Baselines. We compare with Pass@1, fixed-temperature majority voting / self-consistency (Wang et al., 2023; Brown et al., 2024), DeepConf with tail confidence aggregation (Fu et al., 2026b), and TURN as a temperature-selection baseline (Du et al., 2025). These baselines cover single-sample accuracy, answer-frequency aggregation, confidence-aware trace selection, and temperature optimization. DeepConf is filtered by $\eta = 0.9$ and evaluated at its default single-temperature setting since the original method does not perform temperature search. Mean and bottom-10% TN-MAD aggregation variants are evaluated in Section 5.3.

Metrics. We report final-answer accuracy (%) after answer extraction. Table 1 evaluates our default method: TN-MAD tail aggregation, top-η filtering with $\eta = 0.9$, and scaled majority voting as in Equation 6. The same default score aggregation and retention fraction are used for `Ours+TURN`. Section 5.3 isolates the trace-score aggregation choice; Appendix C reports sensitivity to η, the voting-weight transformation, and further diagnostics.

5.2 Main Results

This experiment evaluates TN-MAD in its default cross-temperature inference instantiation against fixed-temperature sampling, majority voting, DeepConf-style confidence selection, and TURN with majority voting using the protocol above. Appendix A.2 reports the additional temperature-selection overhead incurred by TURN.

Table 1 shows the default TN-MAD pipeline without selecting a score-aggregation variant or retention fraction per dataset. `Ours` obtains or ties the best row value in six of the nine model-dataset settings, including clear gains on HMMT25 and BRUMO25 for Qwen3-4B-Thinking-2507 and Qwen3-8B. Since these results combine candidate diversity, scoring, filtering, and voting, Appendix C.4 further isolates the contribution of TN-MAD on identical mixed-temperature pools.

Compatibility with TURN. TN-MAD is a scoring rule and can be incorporated into an existing temperature-selection framework. Under `Ours+TURN`, the TURN-selected temperature T is used as an anchor for a local cross-temperature pool $\{T - 0.1, T, T + 0.1, T + 0.2\}$, followed by the same default tail TN-MAD scoring, $\eta = 0.9$ filtering, and scaled majority voting used by `Ours`. The rightmost comparison in Table 1 shows that this fixed configuration improves over TURN in six of nine settings and ties it in one. These results show that TN-MAD provides a reliable trace-ranking signal for mixed-temperature candidate pools, improving inference performance without dataset-specific selection of aggregation rules or retention settings.

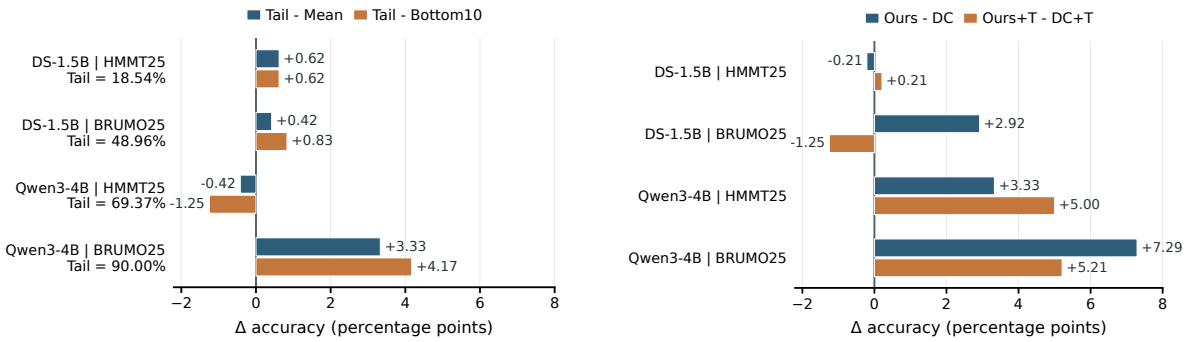
5.3 Ablating Trace Aggregation and Scoring Strategy

This ablation is conducted on four model–benchmark configurations: DeepSeek-R1-Distill-Qwen-1.5B and Qwen3-4B-Thinking-2507 on HMMT25 and BRUMO25. It separates two design choices in the inference pipeline. Panel (a) holds TN-MAD scoring, $\eta = 0.9$ filtering, and scaled majority voting fixed while varying only the trace-level aggregation rule: mean, tail, or bottom-10%. Panel (b) holds tail aggregation and the downstream inference procedure fixed while changing the scoring strategy from DeepConf confidence (`DC`) to TN-MAD, both in the base pipeline and

[1] DeepSeek-R1-Distill-Qwen-1.5B: https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Qwen-1.5B
[2] Qwen3-4B-Thinking-2507: https://huggingface.co/Qwen/Qwen3-4B-Thinking-2507
[3] Qwen3-8B:https://huggingface.co/Qwen/Qwen3-8B

	Comparison methods				TN-MAD variants	
Dataset	Pass@1	Maj	DeepConf	TURN	Ours	Ours+TURN
DeepSeek-R1-Distill-Qwen-1.5B						
AIME25	23.04	37.08	37.08	38.13	**38.33**	**38.33**
HMMT25	14.16	18.33	18.75	22.08	18.54	**22.92**
BRUMO25	29.88	45.21	46.04	**50.42**	48.96	49.17
Qwen3-4B-Thinking-2507						
AIME25	80.73	**86.67**	**86.67**	**86.67**	**86.67**	**86.67**
HMMT25	56.55	63.12	66.04	62.08	**69.37**	67.71
BRUMO25	78.51	81.46	82.71	81.88	**90.00**	89.38
Qwen3-8B						
AIME25	68.54	77.08	77.08	**78.96**	78.75	78.75
HMMT25	44.40	59.79	60.42	60.42	**61.04**	60.62
BRUMO25	69.71	81.25	82.29	82.29	**83.13**	82.71

Table 1: Main accuracy results (%), grouped by full model name. `Ours` uses TN-MAD with default tail aggregation, $\eta = 0.9$, and scaled majority voting; `Ours+TURN` applies the same method to the TURN-anchored local cross-temperature pool. `DeepConf` reports its tail variant with $\eta = 0.9$. Bold and underlined values indicate the best and second-best results in each dataset row.



(a) Aggregation rule: tail relative to alternatives.

(b) Scoring strategy: TN-MAD relative to DeepConf.

Figure 3: Controlled ablations of trace aggregation and scoring strategy (percentage points) at $\eta = 0.9$. Panel (a) reports `Tail - Mean` and `Tail - Bottom10` with TN-MAD held fixed; positive values favor tail aggregation. Panel (b) fixes tail aggregation and reports `Ours - DC` and `Ours+T - DC+T`, where `+T` denotes TURN conditioning; positive values favor TN-MAD scoring. Absolute tail accuracies are printed in panel (a). `DS-1.5B` denotes DeepSeek-R1-Distill-Qwen-1.5B, and `Qwen3-4B` denotes Qwen3-4B-Thinking-2507.

under TURN conditioning. Figure 3 reports the resulting accuracy contrasts in percentage points.

For the aggregation-rule comparison in Figure 3(a), tail outperforms both alternatives in three settings, while bottom-10% is stronger by 1.25 percentage points in one setting. This supports using tail as a fixed principal configuration without claiming that it is uniformly optimal. For the scoring-strategy comparison in panel (b), once aggregation is fixed to tail, TN-MAD exceeds DeepConf in three of four base comparisons, with gains up to 7.29 points, and in three of four TURN-conditioned comparisons, with gains up to 5.21 points. Collectively, this focused ablation confirms that coupling a robust aggregation rule with the temperature-invariant scoring of TN-MAD yields a highly effective and adaptable inference framework.

6 Discussion

Effectiveness and compatibility. TN-MAD addresses a scoring mismatch in cross-temperature sampling: traces generated under different temperature-scaled distributions require a temperature-comparable signal for reliable ranking and aggregation. This scoring mechanism complements existing choices of temperature selection and answer aggregation, allowing it to be incorporated into different inference pipelines. Table 1 evaluates the fixed default instantiation, using tail aggregation and $\eta = 0.9$ both alone and

with TURN anchoring. Section 5.3 isolates the trace-aggregation rule and the scoring strategy, without selecting either configuration per setting. Appendix C reports the remaining retention and voting-rule sensitivity analyses.

Scope of TN-MAD. TN-MAD is designed as a model-internal scoring signal for filtering and voting, rather than as an external verifier. Specifically, it measures the degree of separation among top token candidates under the current reasoning prefix. Unlike raw confidence, which is tied to absolute probability values and may favor conservative high-likelihood traces, TN-MAD instead focuses on centered local separation. This makes it useful for comparing traces across temperatures, but its role is to provide a lightweight reliability signal rather than to independently verify the correctness of a full reasoning trajectory. The score-distribution diagnostics in Appendix D show that correct traces tend to shift toward higher TN-MAD scores across several mixed-temperature pools, which helps explain why top-η filtering can preferentially retain better candidates.

TN-MAD as a plug-in scoring signal. As TN-MAD produces a trace score, it can be used beyond the specific filtering and voting pipeline studied in this paper. Our results show that it can be evaluated in a local temperature neighborhood anchored by TURN and alongside raw-confidence controls. This makes it a candidate plug-in score for scoring-based inference frameworks. For example, in DeepConf-style filtering and weighted-voting pipelines (Fu et al., 2026b), TN-MAD may complement raw confidence scores, especially in TURN-anchored or mixed-temperature candidate pools. Its cross-temperature comparability may also be useful for methods that adjust the sampling temperature during generation, such as entropy-guided temperature control and selective high-temperature sampling (Zhang et al., 2024; Troshin et al., 2025). When such procedures produce traces under different temperature schedules, TN-MAD can provide a common score for comparing and aggregating the completed traces; evaluating this extension is left for future work.

7 Conclusion

This paper introduced the problem of temperature-comparable trace scoring for multi-temperature test-time reasoning and proposed TN-MAD as a solution. Unlike raw probability-based scores, TN-MAD removes the direct temperature factor from local top-k log-probability dispersion, enabling traces sampled at different temperatures to be ranked within a common candidate pool.

We instantiated TN-MAD in a cross-temperature inference pipeline with filtering and scaled majority voting. Across AIME25, HMMT25, and BRUMO25 with three reasoning models, this instantiation remains competitive with fixed-temperature voting, confidence-based selection, and TURN. Compatibility experiments show that TN-MAD can serve as a plug-in score in TURN-anchored and DeepConf-style filtering settings (Fu et al., 2026b).

TN-MAD is not an external verifier and requires access to token-level top-k log probabilities. The current evaluation also uses offline candidate generation rather than an online pruning policy. Future work should study adaptive temperature allocation, online filtering that avoids completing low-value traces, and broader evaluation beyond mathematical reasoning benchmarks.

Limitations

Adaptive temperature intervals. In this work, we mainly use a coarse predefined temperature grid, e.g., $\{0.4, 0.6, 0.8, 1.0\}$, to cover different reasoning regimes. The TURN-combination experiment also uses a simple local neighborhood around the TURN-selected temperature. While these designs are effective and easy to implement, they may still miss useful intermediate temperatures or allocate samples to less effective regions. A more adaptive strategy could learn the interval width, offset, and sample allocation from problem-level or early-generation signals.

Task and model coverage. The evaluation is limited to three mathematical reasoning benchmarks with 30 problems each. These benchmarks are useful stress tests for multi-sample reasoning, but they do not establish that TN-MAD transfers to broader NLP reasoning, open-domain QA, instruction following, or non-mathematical generation tasks. The evaluated models are also from Qwen-derived reasoning families, including DeepSeek-R1-Distill-Qwen and Qwen3 variants. Broader model families and larger benchmark suites are needed before treating TN-MAD as a general test-time scaling score.

White-box probability access. TN-MAD requires access to token-level probability information, especially top-k log probabilities. This makes the method most suitable for white-box or logprob-accessible inference settings. It may be less applicable when only black-box model outputs are available, or when inference APIs do not expose sufficient probability information. Collecting top-k probabilities may also introduce additional engineering overhead in some serving systems.

Cost-performance trade-off. Appendix D reports the token usage of retained traces as a cost proxy, showing that TN-MAD tends to keep shorter traces than raw confidence or no-rank selection under stricter filtering. However, the current evaluation also uses offline candidate generation rather than an online pruning policy. Although TN-MAD may support online decisions such as early stopping or adaptive sample allocation, we do not systematically evaluate such online settings in this work. A full cost-performance study would require an online policy that stops, prunes, or reallocates samples before discarded traces are fully generated.

References

Art of Problem Solving. 2025a. 2025 AIME I. Accessed: 2026-05-23.

Art of Problem Solving. 2025b. 2025 AIME II. Accessed: 2026-05-23.

Sourya Basu, Govardana Sachitanandam Ramachandran, Nitish Shirish Keskar, and Lav R. Varshney. 2021. Mirostat: A neural text decoding algorithm that directly controls perplexity. In *International Conference on Learning Representations*.

Bradley Brown, Jordan Juravsky, Ryan Ehrlich, Ronald Clark, Quoc V. Le, Christopher Ré, and Azalia Mirhoseini. 2024. Large Language Monkeys: Scaling Inference Compute with Repeated Sampling. *Preprint*, arXiv:2407.21787. ArXiv preprint arXiv:2407.21787.

Brown University Math Olympiad. 2025. Brown university math olympiad 2025 problem archive. Accessed: 2026-05-25.

Chengkun Cai, Xu Zhao, Yucheng Du, Haoliang Liu, and Lei Li. 2024. T^2 of Thoughts: Temperature Tree Elicits Reasoning in Large Language Models. *Preprint*, arXiv:2405.14075. ArXiv preprint arXiv:2405.14075.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training Verifiers to Solve Math Word Problems. *Preprint*, arXiv:2110.14168. ArXiv preprint arXiv:2110.14168.

DeepSeek-AI. 2025. DeepSeek-R1-Distill-Qwen-1.5B. Model card. Accessed: 2026-05-23.

Weihua Du, Yiming Yang, and Sean Welleck. 2025. Optimizing temperature for language models with multi-sample inference. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 14648–14668. PMLR.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical Neural Story Generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898, Melbourne, Australia. Association for Computational Linguistics.

Yichao Fu, Junda Chen, Siqi Zhu, Zheyu Fu, Zhongdongming Dai, Yonghao Zhuang, Yian Ma, Aurick Qiao, Tajana Rosing, Ion Stoica, and Hao Zhang. 2026a. Efficiently scaling LLM reasoning programs with certaindex. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Yichao Fu, Xuewei Wang, Hao Zhang, Yuandong Tian, and Jiawei Zhao. 2026b. Deep think with confidence. In *The Fourteenth International Conference on Learning Representations*.

Jiahui Geng, Fengyu Cai, Yuxia Wang, Heinz Koeppl, Preslav Nakov, and Iryna Gurevych. 2024. A survey of confidence estimation and calibration in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6577–6595, Mexico City, Mexico. Association for Computational Linguistics.

Harvard-MIT Mathematics Tournament. 2025. February 2025. Tests and solutions archive. Accessed: 2026-05-25.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

Zhewei Kang, Xuandong Zhao, and Dawn Song. 2025. Scalable best-of-n selection for large language models via self-certainty. In *Advances in Neural Information Processing Systems*.

Xueyan Li, Guinan Su, Mrinmaya Sachan, and Jonas Geiping. 2026. Sample smart, not hard: Correctness-first decoding for better reasoning in LLMs. In *The Fourteenth International Conference on Learning Representations*.

Nguyen Nhat Minh, Andrew Baker, Clement Neo, Allen G Roush, Andreas Kirsch, and Ravid Shwartz-Ziv. 2025. Turning up the heat: Min-p sampling for creative and coherent LLM outputs. In *The Thirteenth International Conference on Learning Representations*.

Qwen Team. 2025a. Qwen3-4B-Thinking-2507. Model card. Accessed: 2026-05-23.

Qwen Team. 2025b. Qwen3-8B. Model card. Accessed: 2026-05-23.

Mousa Salah and Amgad Muneer. 2026. Temperature-Dependent Performance of Prompting Strategies in Extended Reasoning Large Language Models. *Preprint*, arXiv:2604.08563. ArXiv preprint arXiv:2604.08563.

Yung-Chen Tang, Pin-Yu Chen, and Andrea Cavallaro. 2025. CarBoN: Calibrated Best-of-N Sampling Improves Test-time Reasoning. *Preprint*, arXiv:2510.15674. ArXiv preprint arXiv:2510.15674.

Sergey Troshin, Wafaa Mohammed, Yan Meng, Christof Monz, Antske Fokkens, and Vlad Niculae. 2025. Control the temperature: Selective sampling for diverse and high-quality LLM outputs. In *Second Conference on Language Modeling*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

Sean Welleck, Amanda Bertsch, Matthew Finlayson, Hailey Schoelkopf, Alex Xie, Graham Neubig, Ilia Kulikov, and Zaid Harchaoui. 2024. From decoding to meta-generation: Inference-time algorithms for large language models. *Transactions on Machine Learning Research*.

Yuheng Wu and Thierry Tambe. 2025. On the role of temperature sampling in test-time scaling. In *NeurIPS 2025 Workshop on Foundations of Reasoning in Language Models*.

Ziang Xie. 2017. Neural Text Generation: A Practical Guide. Technical report.

Shimao Zhang, Yu Bao, and Shujian Huang. 2024. EDT: Improving Large Language Models' Generation by Entropy-based Dynamic Temperature Sampling. *Preprint*, arXiv:2403.14541. ArXiv preprint arXiv:2403.14541.

Yuqi Zhu, Jia Li, Ge Li, YunFei Zhao, Jia Li, Zhi Jin, and Hong Mei. 2024. Hot or cold? adaptive temperature sampling for code generation with large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 437–445.

Appendix

A Experimental Configuration Details

A.1 Experiment Configuration

Candidate-pool hyperparameters. For each benchmark, we evaluate 30 problems. The generation stage stores 128 traces for each problem-temperature pair. Unless otherwise stated, the main cross-temperature grid is $\{0.4, 0.6, 0.8, 1.0\}$. During evaluation, each cross-temperature working set contains 64 traces by sampling 16 traces from each temperature, while each fixed-temperature working set contains 64 traces from a single temperature. All single-temperature baselines use the recommended temperature $T = 0.6$ for the three evaluated models (DeepSeek-AI, 2025; Qwen Team, 2025a,b). Reported values are averaged over 16 repeated subsampling runs.

Generation hyperparameters. We generate candidate traces with vLLM and use each model's native tokenizer. The implementation explicitly controls the sampling temperature and records the top-20 token log probabilities at each generation step. Following the top-k notation in Section 4, all TN-MAD scores in the experiments are computed with $k = 20$. The same log-probabilities are also used by confidence-based scores. Special tokens are retained in the generated output. Decoding terminates when the model emits an end-of-sequence token or when the maximum generation length is reached.

The maximum generation length is model-specific: DeepSeek-R1-Distill-Qwen-1.5B uses 64000 tokens, Qwen3-4B-Thinking-2507 uses 128000 tokens, and Qwen3-8B uses 38912 tokens. Qwen3-8B is evaluated with thinking mode enabled and `presence_penalty=1.0`. Other sampling controls use the model defaults unless explicitly stated.

Prompt templates. For all models, we append the same instruction to each problem prompt: "Please reason step by step, and put your final answer within `\boxed{}`." If a tokenizer chat template is available, the prompt is wrapped with that template and a generation prompt is added. Qwen3-8B is evaluated with thinking mode enabled. For DeepSeek-R1-Distill-Qwen-1.5B, the script appends `\<think>` after the chat template. In all cases, the final answer is expected to appear inside `\boxed{}`.

Model	AIME25	HMMT25	BRUMO25
DeepSeek-R1-Distill-Qwen-1.5B	0.5	0.2	0.2
Qwen3-4B-Thinking-2507	0.2	0.2	0.3
Qwen3-8B	0.4	0.2	0.4

Table A.1: TURN-anchor temperatures by model. Each entry is the temperature anchor used for the corresponding model-dataset pair.

TURN Temperature Anchors. We follow the default hyperparameter settings documented by TURN (Du et al., 2025). Specifically, under the default settings, the program sweeps the temperature range from 0.1 to 1.5 with a step size of 0.1, limits the generation for each question to at most 1024 tokens, and samples a total of 32 responses. During generation, the top 1000 logprobs at each step are extracted to accurately compute the entropy of the generated text. In addition, our settings for the problem prompt and chat template are consistent with those used in the main experiments.

Table A.1 reports the TURN-anchor temperatures used in the TURN baseline and in TURN-anchored combinations. For `Ours+TURN`, the anchor temperature T defines the local candidate temperature pool described in the main text.

A.2 Budget Accounting

Table A.2 separates the final candidate pool used for accuracy evaluation from any additional temperature-selection stage. Throughout the main comparison, matching applies to the pool aggregated into an answer, rather than to total generation cost.

Protocol interpretation and token-cap accounting. TURN-based settings first execute the official temperature-selection procedure, generating 32 short responses per problem capped at 1,024 tokens each, before constructing the final pool summarized in Table A.2. The default cross-temperature TN-MAD protocol has no separate temperature-selection stage. Comparisons involving TURN therefore evaluate final-pool effectiveness while acknowledging protocol-specific selection overhead. The main experiments record top-20 log probabilities per generated token and use model-specific maximum generation lengths; these caps are upper bounds rather than observed averages. A compute-matched efficiency study would require observed generated-token or latency measurements, including the cost of token-level log-probability extraction used by TN-MAD.

Method	Role	Pool used for reported accuracy	Generated responses per test question			
			Search / select	Final pool	Total	Ratio to ours
Fixed $T = 0.6$	Deployable baseline	64 traces at $T = 0.6$	0	64	64	$1.0\times$
Dev-tuned single T	Calibrated baseline	64 traces at a held-out selected T	Offline[†]	64	64[†]	$1.0\times$[†]
Best single T (hindsight)	Oracle only	Best observed 64-trace pool across four T values	4×64[‡]	64 reused	256	$4.0\times$
TURN	Selection baseline	64 traces at the TURN-selected T	32 short[§]	64	96	$1.5\times$[§]
Cross-temp. TN-MAD	Proposed default	16 traces at each $T \in \{0.4, 0.6, 0.8, 1.0\}$	0	64	64	$1.0\times$
TN-MAD + TURN	Anchored variant	64 traces in a local pool around the TURN anchor	32 short[§]	64	96	$1.5\times$[§]

Table A.2: Response-count accounting for protocols compared through a 64-trace final answer pool. [†]Held-out tuning is offline and is excluded from the per-question online total when the selected temperature is reused. [‡]The hindsight oracle generates four complete pools and reuses the selected pool for aggregation; it is not deployable. [§]TURN selection generates 32 short responses capped at 1,024 tokens each. Ratios compare response counts only and do not establish matched token cost or latency.

B Temperature Sensitivity of Raw Scores

This appendix provides a theoretical note on why raw confidence and entropy scores are sensitive to temperature, and why TN-MAD removes the direct temperature factor. We focus on one decoding step t. Let $z_t(v)$ denote the logit of vocabulary token $v \in \mathcal{V}$, and let $\{z_{t,1}, z_{t,2}, \ldots, z_{t,k}\}$ denote the logits of the returned top-k tokens at this step. Throughout this appendix, top-k probabilities are top-k entries from the full-softmax distribution, not probabilities re-normalized within the top-k set:

$$p_{t,j}(T) = \frac{\exp(z_{t,j}/T)}{\sum_{v\in\mathcal{V}} \exp(z_t(v)/T)}, \quad j = 1, \ldots, k.$$

Let

$$Z_t(T) = \sum_{v\in\mathcal{V}} \exp(z_t(v)/T).$$

Then

$$\log p_{t,j}(T) = \frac{z_{t,j}}{T} - \log Z_t(T).$$

For a fixed local logit vector, a temperature-comparable score should not change its value or ranking solely because the same logits are rescaled by temperature. For this analysis, we use **temperature-unified** to mean the value of a score for a fixed logit vector is independent of T, and **temperature-ranking invariant** to mean the ranking between two fixed logit vectors is independent of T.

B.1 Temperature Dependence and Ranking-Change Risk in Raw Scores

Raw confidence and entropy both depend on the temperature-scaled probability distribution. Their values and rankings can therefore change when the same logits are evaluated under different temperatures.

For raw top-k confidence,

$$C_t(T) = -\frac{1}{k}\sum_{j=1}^{k} \log p_{t,j}(T).$$

Substituting the temperature-scaled log-probability gives

$$\begin{aligned} C_t(T) &= -\frac{1}{k}\sum_{j=1}^{k}\left(\frac{z_{t,j}}{T} - \log Z_t(T)\right) \\ &= \log Z_t(T) - \frac{\bar{z}_t}{T}. \end{aligned} \tag{8}$$

where

$$\bar{z}_t = \frac{1}{k}\sum_{j=1}^{k} z_{t,j}.$$

Equation 8 shows that $C_t(T)$ depends on T. Therefore, raw confidence is not temperature-unified.

The temperature dependence can be made explicit by considering two limits. Let $z^{\mathcal{V}}_{t,\max} =$

$\max_{v\in\mathcal{V}} z_t(v)$. When $T \to 0^+$, assuming the maximum vocabulary logit is unique,

$$\log Z_t(T) = \frac{z^{\mathcal{V}}_{t,\max}}{T} + o(1).$$

Thus,

$$C_t(T) = \frac{z^{\mathcal{V}}_{t,\max} - \bar{z}_t}{T} + o(1), \quad T \to 0^+.$$

When $T \to +\infty$, set $\beta = 1/T$. By expanding $\log Z_t(T) = \log \sum_{v\in\mathcal{V}} \exp(\beta z_t(v))$ around $\beta = 0$, we obtain

$$\log Z_t(T) = \log|\mathcal{V}| + \frac{\mu^{\mathcal{V}}_t}{T} + \frac{\mathrm{Var}_{\mathcal{V}}(z_t)}{2T^2} + O(T^{-3}),$$

where

$$\mu^{\mathcal{V}}_t = \frac{1}{|\mathcal{V}|}\sum_{v\in\mathcal{V}} z_t(v),$$
$$\mathrm{Var}_{\mathcal{V}}(z_t) = \frac{1}{|\mathcal{V}|}\sum_{v\in\mathcal{V}} (z_t(v) - \mu^{\mathcal{V}}_t)^2.$$

Therefore,

$$C_t(T) = \log|\mathcal{V}| + \frac{\mu^{\mathcal{V}}_t - \bar{z}_t}{T} + \frac{\mathrm{Var}_{\mathcal{V}}(z_t)}{2T^2} + O(T^{-3}), \qquad T \to +\infty. \tag{9}$$

These two limits show that raw confidence is dominated by different logit statistics under different temperatures: $z^{\mathcal{V}}_{t,\max} - \bar{z}_t$ at low temperature and vocabulary-level terms such as $\mu^{\mathcal{V}}_t - \bar{z}_t$ and $\mathrm{Var}_{\mathcal{V}}(z_t)$ at high temperature. Since these statistics do not induce the same ordering over local top-k logit patterns in general, raw confidence can change its ranking under temperature scaling. Thus, raw confidence is not temperature-ranking invariant.

The same issue also appears for top-k entropy:

$$H_t(T) = -\sum_{j=1}^{k} p_{t,j}(T)\log p_{t,j}(T).$$

Using $\log p_{t,j}(T) = z_{t,j}/T - \log Z_t(T)$, and writing $P_{t,k}(T) = \sum_{j=1}^{k} p_{t,j}(T)$ for the full-softmax mass covered by the returned top-k tokens, top-k entropy becomes

$$H_t(T) = P_{t,k}(T)\log Z_t(T) - \frac{1}{T}\sum_{j=1}^{k} p_{t,j}(T) z_{t,j}.$$

Therefore, raw entropy is directly temperature-dependent. Its ranking behavior also changes with temperature.

For the low-temperature limit, assume the maximum vocabulary logit is unique and belongs to the returned top-k set. Let $j^\star$ index this maximum among the top-k entries. The full-softmax mass then concentrates on $j^\star$, and the leading behavior is controlled by the gaps between the maximum logit and the nearest competing logits:

$$\Delta_v = z_{t,j^\star} - z_t(v), \quad v \neq j^\star.$$

At low temperature, smaller gaps produce a less concentrated local distribution and a larger entropy contribution from high-probability alternatives. Thus, low-temperature entropy ranking is mainly controlled by top-logit separation.

For the high-temperature limit, the full-softmax distribution approaches the uniform distribution over $\mathcal{V}$, so the returned top-k mass approaches $k/|\mathcal{V}|$. Consequently, the top-k entropy approaches a vocabulary-normalized constant rather than the entropy of a top-k re-normalized distribution:

$$H_t(T) = \frac{k}{|\mathcal{V}|}\log|\mathcal{V}| + O(T^{-1}), \quad T \to +\infty.$$

The correction terms depend on the top-k logits, the vocabulary-level normalization, and their interaction through $P_{t,k}(T)$. Thus, high-temperature entropy is controlled by a different set of statistics from the low-temperature top-gap behavior.

Because low-temperature entropy depends primarily on the top-logit gap, while high-temperature entropy depends on the full-softmax mass and vocabulary-level normalization, entropy can also change its ranking under temperature scaling. Therefore, raw entropy is not temperature-ranking invariant.

Together, these two cases show a theoretical score-ranking inversion risk of raw scores in a cross-temperature candidate pool: the same local logits can receive different score rankings solely because the scoring function responds to temperature-induced reshaping of the probability distribution.

B.2 TN-MAD Is Temperature-Ranking Invariant

TN-MAD is defined as

$$M_t(T) = \frac{T}{k}\sum_{j=1}^{k}\left|\log p_{t,j}(T) - \overline{\log p_t(T)}\right|,$$

where

$$\overline{\log p_t(T)} = \frac{1}{k}\sum_{j=1}^{k} \log p_{t,j}(T).$$

Using $\log p_{t,j}(T) = z_{t,j}/T - \log Z_t(T)$, we have

$$\overline{\log p_t(T)} = \frac{\bar{z}_t}{T} - \log Z_t(T).$$

Therefore, the shared normalization term cancels:

$$\log p_{t,j}(T) - \overline{\log p_t(T)} = \frac{z_{t,j} - \bar{z}_t}{T}.$$

Substituting this into TN-MAD gives

$$M_t(T) = \frac{T}{k}\sum_{j=1}^{k}\left|\frac{z_{t,j} - \bar{z}_t}{T}\right| = \frac{1}{k}\sum_{j=1}^{k}|z_{t,j} - \bar{z}_t|.$$

This recovers the temperature-free form in Equation 4. The right-hand side contains no temperature term. Therefore, for a fixed top-k logit vector, TN-MAD is temperature-unified.

For any two fixed logit vectors $z^{(a)}$ and $z^{(b)}$, define

$$D(z) = \frac{1}{k}\sum_{j=1}^{k}|z_j - \bar{z}|.$$

Since $M^{(a)}(T) = D(z^{(a)})$ and $M^{(b)}(T) = D(z^{(b)})$ for all $T > 0$, we have

$$M^{(a)}(T_0) > M^{(b)}(T_0) \iff M^{(a)}(T) > M^{(b)}(T), \quad \forall T, T_0 > 0.$$

This establishes that TN-MAD is temperature-ranking invariant under the same top-k candidate set and full-softmax log-probabilities.

B.3 Extension to Token-Wise Temperature

The same argument applies when temperature changes inside a trace. Suppose token t is generated with token-wise temperature T_t. Then

$$p_{t,j}(T_t) = \frac{\exp(z_{t,j}/T_t)}{\sum_{v\in\mathcal{V}}\exp(z_t(v)/T_t)}.$$

TN-MAD becomes

$$M_t(T_t) = \frac{T_t}{k}\sum_{j=1}^{k}\left|\log p_{t,j}(T_t) - \overline{\log p_t(T_t)}\right|.$$

Since

$$\log p_{t,j}(T_t) - \overline{\log p_t(T_t)} = \frac{z_{t,j} - \bar{z}_t}{T_t},$$

we again obtain

$$M_t(T_t) = \frac{1}{k}\sum_{j=1}^{k}|z_{t,j} - \bar{z}_t|.$$

Thus, fixed-temperature decoding is the special case $T_t = T$, and the trace-level aggregation formulas remain valid by replacing $M_t(T)$ with $M_t(T_t)$ at each token position.

C Ablation

C.1 Cross-Temperature Ablation

We evaluate whether a TN-MAD-based cross-temperature instantiation can reduce the need for temperature-specific tuning. As a non-default score-aggregation diagnostic, this comparison uses mean TN-MAD aggregation, a retention fraction of $\eta = 0.9$, and scaled majority voting, with baselines evaluated at fixed temperatures. Each baseline is allowed to select its best single-temperature result. The main-table default instead uses tail aggregation.

Table C.1 should not be read as a directly available inference baseline, but an oracle comparison. The temperature reported in "Best Single Temp" is unknown before evaluation for a new model-dataset setting. Reproducing this oracle by brute-force search over $\{0.4, 0.6, 0.8, 1.0\}$ would require more budget than the TN-MAD-based mixed-temperature setting. Appendix A.2 gives the corresponding response-budget accounting and separates this oracle search cost from the answer-aggregation pool used for accuracy reporting.

This budget-aware comparison shows that the TN-MAD-based instantiation approaches the benefit of the unknown best temperature without first performing an exhaustive temperature sweep. With one quarter of the candidate-generation budget required by that oracle search, the $\eta = 0.9$ cross-temperature configuration is consistently stronger than single-temperature Pass@1 and remains in the same performance range as strong single-temperature aggregation references. In several settings it matches or exceeds those hindsight references, indicating that TN-MAD filtering and scaled majority voting can use complementary traces from different temperatures effectively.

C.2 Top-η Retention Ablation

We next examine how much of the sampled candidate pool should be retained before voting. This

ablation fixes TN-MAD with MajScale and varies η for tail aggregation (the main-table default) as well as the mean and bottom-10% aggregation controls. The purpose is to show how sensitive the method is to candidate filtering, rather than retroactively tuning the main experimental setting.

Table C.2 summarizes the trend across all model, dataset, score-source, and TURN conditions. Very aggressive filtering is consistently harmful: retaining only the top 10% of traces lowers the average accuracy across all conditions. Once at least half of the pool is retained, performance becomes much more stable. The best aggregate value occurs around mild filtering, with $\eta = 0.50$ reaching 63.83 average accuracy across all 54 conditions.

Tables C.3 and C.4 show why there is no universally dominant retention fraction at the condition level. Some settings benefit from moderate filtering, especially when low-scoring traces add noise to the vote. Other settings, such as several AIME25 cases, prefer no filtering or tie with no filtering because the retained pool is already small or the candidate answers are highly concentrated. The TURN conditions follow the same broad pattern but are slightly more favorable to high-retention settings, which is consistent with TURN already providing a useful temperature anchor before TN-MAD ranks the local candidate pool.

We therefore use $\eta = 0.90$ as a conservative mild-filtering choice rather than as a per-setting tuned optimum. It avoids the clear degradation caused by overly strict filtering, remains close to the best aggregate region, and is compatible with the top-90% filtering convention used in DeepConf-style analyses (Fu et al., 2026b). The main claim supported by this ablation is not that $\eta = 0.90$ is optimal for every model-dataset pair, but that TN-MAD is robust in the high-retention regime where most candidate traces remain available for aggregation.

C.3 MajWeight vs. MajScale Ablation

We also isolate the effect of the voting rule after candidate ranking and retention have been fixed. Direct score-weighted voting, denoted MajWeight, uses the TN-MAD scores themselves as voting weights. In contrast, MajScale first min-max normalizes the retained scores within each problem and then applies the scaled weights in Equations (6) and (7). This comparison tests whether within-problem score normalization helps aggregation beyond the effect of top-η filtering.

Table C.5 reports both the overall comparison and the stratified differences. The pattern depends strongly on the retention regime. When filtering is strict, MajWeight performs better: at $\eta = 0.10$ and $\eta = 0.25$, the retained set is small, and direct weighting preserves the stronger separation among the few selected traces. As η increases, the advantage shifts to MajScale. From $\eta = 0.50$ onward, MajScale has the higher mean accuracy, with the largest average gain appearing at $\eta = 0.75$.

The stratified panel shows that this trend is not driven by only one setting. Positive MajScale-minus-MajWeight differences appear for both Base and TURN conditions once $\eta \geq 0.50$, and the same direction holds across Mean, Tail, and Bottom10 score sources. This suggests that score scaling is most useful when the retained pool is large enough for many traces to participate in the vote. In that regime, raw score magnitudes can overemphasize a small number of traces, while min-max scaling preserves the ranking signal but produces a more balanced vote over the retained candidates.

This ablation explains why the main compatibility experiment uses MajScale in the mild-filtering regime. MajScale is not uniformly superior under very aggressive filtering, but it is the more reliable aggregation choice when the method keeps most of the cross-temperature pool, especially around $\eta = 0.90$.

C.4 Same-Pool Scorer Attribution

The main experiments combine three effects: mixed-temperature coverage, TN-MAD scoring, and weighted answer aggregation. We therefore run an additional same-pool attribution analysis that fixes the exact mixed-temperature pools before comparing scoring rules. For each problem, we form eight disjoint mixed pools by taking 16 traces from each temperature in $\{0.4, 0.6, 0.8, 1.0\}$, and all compared mixed-pool methods use the same pool manifest. The statistical unit is the problem, and intervals are paired bootstrap intervals over problem-level outcomes.

Table C.6 expands the attribution analysis with available scoring controls. `RawConf` refers to raw confidence directly calculated from probabilities from which the token is generated. `TNormConf` uses the original logits, which equal the normalized form at $T = 1$. TN-MAD improves over unweighted mixed-pool majority voting by +0.93 percentage points and over raw confidence by +1.02 points without filtering. However, it is not

			Fixed single-temperature accuracy				Cross-temperature comparison		
Model	Dataset	Baseline	$T = 0.4$	$T = 0.6$	$T = 0.8$	$T = 1.0$	Best single T	Ours	Δ
1.5B	AIME25	Pass@1	22.15	23.04	24.32	23.71	0.8 (24.32)	38.12	+13.80
		Maj	37.08	37.08	40.00	38.54	0.8 (40.00)	38.12	-1.88
		DeepConf-Tail	36.87	37.08	40.63	38.96	0.8 (40.63)	38.12	-2.50
		DeepConf-Bottom10	37.08	37.29	39.79	37.92	0.8 (39.79)	38.12	-1.67
1.5B	HMMT25	Pass@1	12.66	14.16	13.77	13.42	0.6 (14.16)	17.92	+3.76
		Maj	17.08	18.33	17.50	18.96	1.0 (18.96)	17.92	-1.04
		DeepConf-Tail	17.50	18.75	17.92	20.21	1.0 (20.21)	17.92	-2.29
		DeepConf-Bottom10	17.08	18.54	18.33	19.17	1.0 (19.17)	17.92	-1.25
1.5B	BRUMO25	Pass@1	25.95	29.88	31.74	30.63	0.8 (31.74)	48.54	+16.80
		Maj	47.92	45.21	45.42	47.71	0.4 (47.92)	48.54	+0.63
		DeepConf-Tail	49.38	46.04	48.96	48.33	0.4 (49.38)	48.54	-0.83
		DeepConf-Bottom10	47.29	46.46	44.58	48.33	1.0 (48.33)	48.54	+0.21
4B	AIME25	Pass@1	80.76	80.73	80.66	80.29	0.4 (80.76)	86.67	+5.91
		Maj	86.67	86.67	86.67	86.67	0.4 (86.67)	86.67	0.00
		DeepConf-Tail	86.67	86.67	86.67	86.67	0.4 (86.67)	86.67	0.00
		DeepConf-Bottom10	86.67	86.67	86.67	86.67	0.4 (86.67)	86.67	0.00
4B	HMMT25	Pass@1	56.46	56.55	55.85	55.59	0.6 (56.55)	69.79	+13.24
		Maj	64.58	63.12	59.79	62.50	0.4 (64.58)	69.79	+5.21
		DeepConf-Tail	64.17	66.04	64.17	63.96	0.6 (66.04)	69.79	+3.75
		DeepConf-Bottom10	65.21	64.79	63.96	63.75	0.4 (65.21)	69.79	+4.58
4B	BRUMO25	Pass@1	79.71	78.51	79.72	77.71	0.8 (79.72)	86.67	+6.95
		Maj	83.75	81.46	83.54	82.71	0.4 (83.75)	86.67	+2.92
		DeepConf-Tail	85.21	82.71	85.63	84.17	0.8 (85.63)	86.67	+1.04
		DeepConf-Bottom10	84.38	82.29	83.96	82.71	0.4 (84.38)	86.67	+2.29
8B	AIME25	Pass@1	67.59	68.54	68.35	69.57	1.0 (69.57)	78.54	+8.97
		Maj	78.96	77.08	77.29	83.33	1.0 (83.33)	78.54	-4.79
		DeepConf-Tail	79.38	77.08	76.88	83.33	1.0 (83.33)	78.54	-4.79
		DeepConf-Bottom10	79.79	76.67	78.54	83.33	1.0 (83.33)	78.54	-4.79
8B	HMMT25	Pass@1	43.47	44.40	44.99	45.84	1.0 (45.84)	60.42	+14.57
		Maj	59.17	59.79	59.79	58.75	0.6 (59.79)	60.42	+0.62
		DeepConf-Tail	59.79	60.42	60.00	59.17	0.6 (60.42)	60.42	0.00
		DeepConf-Bottom10	59.79	60.21	60.00	59.79	0.6 (60.21)	60.42	+0.21
8B	BRUMO25	Pass@1	69.49	69.71	70.31	69.48	0.8 (70.31)	81.67	+11.36
		Maj	82.29	81.25	81.04	80.63	0.4 (82.29)	81.67	-0.62
		DeepConf-Tail	82.50	82.29	82.08	80.63	0.4 (82.50)	81.67	-0.83
		DeepConf-Bottom10	82.29	81.88	82.71	79.79	0.8 (82.71)	81.67	-1.04

Table C.1: Cross-temperature ablation against single-temperature baselines. Fixed-temperature columns report results at each temperature. The best single-temperature result is selected in hindsight. Δ denotes Ours minus the best single-temperature result.

		Retained fraction						Selected setting	
Setting	N	$\eta = 0.10$	$\eta = 0.25$	$\eta = 0.50$	$\eta = 0.75$	$\eta = 0.90$	$\eta = 1.00$	Best η	Gain
Base	27	59.65	63.17	63.65	63.38	63.40	63.03	0.50 (63.65)	+0.62
TURN	27	59.39	62.98	64.00	64.18	63.90	63.46	0.75 (64.18)	+0.72
All	54	59.52	63.07	63.83	63.78	63.65	63.25	0.50 (63.83)	+0.58

Table C.2: Top-η retention ablation: aggregate trend across Base and TURN conditions. Gain is measured against no filtering, i.e., $\eta = 1.00$.

clearly separated from `TNormConf`; relative to the better of `RawConf` and `TNormConf` on each problem, the full TN-MAD variant differs by +0.28 pp (95% CI [-0.19, 0.74]). Its trace-level AUROC is higher than reconstructed raw confidence (0.804 vs. 0.590 macro average) but only modestly higher than `TNormConf` (0.789), so the evidence supports raw-confidence correction and a useful scoring signal rather than dominance over temperature-normalized alternatives.

C.5 Merged-Pool Factor Ablation

We also test whether TN-MAD remains useful when the candidate pool is expanded before filtering. This diagnostic merges four single-temperature 64-trace pools into 256 traces and retains the top 25% for voting. Here the confidence control is raw confidence.

Table C.8 shows that, in this merged-pool diagnostic, TN-MAD exceeds raw confidence by +2.27 percentage points on average and achieves higher trace-level AUROC (0.80 vs. 0.59). This comparison supports TN-MAD against unadjusted confidence in a genuinely mixed-temperature pool. It does not contradict Table C.6, where temperature-normalized confidence remains competitive.

			Retained fraction						Selected setting	
Model	Dataset	Source	$\eta = .10$	$\eta = .25$	$\eta = .50$	$\eta = .75$	$\eta = .90$	$\eta = 1.00$	Best η	Gain
1.5B	AIME25	Mean	25.42	34.17	37.71	37.92	38.12	38.12	.90/1.00 (38.12)	+0.00
		Tail	32.29	38.12	38.33	38.54	38.33	38.33	.75 (38.54)	+0.21
		Bottom10	31.67	36.46	38.12	37.71	38.33	38.75	1.00 (38.75)	+0.00
1.5B	HMMT25	Mean	16.67	18.12	19.38	18.33	17.92	17.92	.50 (19.38)	+1.46
		Tail	18.96	21.04	21.67	20.21	18.54	18.54	.50 (21.67)	+3.13
		Bottom10	17.08	18.12	18.33	18.33	17.92	17.92	.50/.75 (18.33)	+0.42
1.5B	BRUMO25	Mean	32.29	44.79	47.29	48.33	48.54	48.33	.90 (48.54)	+0.21
		Tail	42.08	47.92	49.17	49.58	48.96	48.96	.75 (49.58)	+0.62
		Bottom10	42.50	45.42	47.71	47.92	48.13	47.92	.90 (48.13)	+0.21
4B	AIME25	Mean	85.42	86.67	86.67	86.67	86.67	86.67	.25/.50/.75/.90/1.00 (86.67)	+0.00
		Tail	82.92	86.67	86.67	86.67	86.67	86.67	.25/.50/.75/.90/1.00 (86.67)	+0.00
		Bottom10	86.04	86.67	86.67	86.67	86.67	86.67	.25/.50/.75/.90/1.00 (86.67)	+0.00
4B	HMMT25	Mean	66.87	70.42	70.42	70.21	69.79	68.54	.25/.50 (70.42)	+1.88
		Tail	64.79	67.50	68.33	68.75	69.37	68.54	.90 (69.37)	+0.83
		Bottom10	68.75	72.29	71.87	70.42	70.62	68.12	.25 (72.29)	+4.17
4B	BRUMO25	Mean	88.33	90.63	87.29	85.63	86.67	85.42	.25 (90.63)	+5.21
		Tail	86.25	89.17	90.00	88.33	90.00	87.50	.50/.90 (90.00)	+2.50
		Bottom10	88.75	92.08	87.50	86.04	85.83	85.00	.25 (92.08)	+7.08
8B	AIME25	Mean	73.96	76.46	77.08	78.13	78.54	79.17	1.00 (79.17)	+0.00
		Tail	74.58	77.50	78.96	78.96	78.75	79.38	1.00 (79.38)	+0.00
		Bottom10	75.21	77.50	78.54	78.75	79.17	79.58	1.00 (79.58)	+0.00
8B	HMMT25	Mean	59.58	60.42	61.25	61.04	60.42	60.21	.50 (61.25)	+1.04
		Tail	54.79	61.87	61.87	61.46	61.04	60.21	.25/.50 (61.87)	+1.67
		Bottom10	56.46	61.04	60.83	60.83	60.21	59.79	.25 (61.04)	+1.25
8B	BRUMO25	Mean	79.58	80.63	82.08	81.46	81.67	81.46	.50 (82.08)	+0.62
		Tail	80.63	82.50	82.08	82.08	83.13	82.50	.90 (83.13)	+0.63
		Bottom10	78.75	81.46	82.71	82.29	81.88	81.67	.50 (82.71)	+1.04

Table C.3: Top-η retention ablation for the Base cross-temperature pool. Gain is measured against no filtering, i.e., $\eta = 1.00$.

			Retained fraction						Selected setting	
Model	Dataset	Source	$\eta = .10$	$\eta = .25$	$\eta = .50$	$\eta = .75$	$\eta = .90$	$\eta = 1.00$	Best η	Gain
1.5B	AIME25	Mean	21.88	29.79	35.83	37.08	37.08	37.92	1.00 (37.92)	+0.00
		Tail	32.71	33.96	37.50	37.71	38.33	38.33	.90/1.00 (38.33)	+0.00
		Bottom10	29.17	34.79	36.87	37.29	38.13	38.13	.90/1.00 (38.13)	+0.00
1.5B	HMMT25	Mean	17.92	24.38	24.58	24.58	24.58	24.38	.50/.75/.90 (24.58)	+0.21
		Tail	16.25	22.92	22.50	22.50	22.92	22.92	.25/.90/1.00 (22.92)	+0.00
		Bottom10	19.17	22.08	21.67	21.67	21.67	21.67	.25 (22.08)	+0.42
1.5B	BRUMO25	Mean	40.83	51.04	52.08	52.08	51.04	51.46	.50/.75 (52.08)	+0.62
		Tail	39.58	48.12	50.00	50.00	49.17	49.38	.50/.75 (50.00)	+0.62
		Bottom10	43.75	48.13	50.83	51.25	50.42	50.62	.75 (51.25)	+0.62
4B	AIME25	Mean	82.08	85.83	86.67	86.67	86.67	86.67	.50/.75/.90/1.00 (86.67)	+0.00
		Tail	86.25	86.88	86.67	86.67	86.67	86.67	.25 (86.88)	+0.21
		Bottom10	84.38	86.67	86.67	86.67	86.67	86.67	.25/.50/.75/.90/1.00 (86.67)	+0.00
4B	HMMT25	Mean	66.67	69.58	70.00	71.04	69.58	67.29	.75 (71.04)	+3.75
		Tail	62.92	65.62	67.71	69.37	67.71	66.04	.75 (69.37)	+3.33
		Bottom10	70.42	72.71	71.67	70.00	68.54	65.00	.25 (72.71)	+7.71
4B	BRUMO25	Mean	81.67	85.83	88.33	86.88	85.21	84.17	.50 (88.33)	+4.17
		Tail	83.54	87.50	89.58	89.79	89.38	86.88	.75 (89.79)	+2.92
		Bottom10	88.75	90.00	89.58	88.54	87.92	84.79	.25 (90.00)	+5.21
8B	AIME25	Mean	75.63	77.50	77.92	78.54	78.96	79.58	1.00 (79.58)	+0.00
		Tail	71.87	77.08	77.71	78.33	78.75	79.17	1.00 (79.17)	+0.00
		Bottom10	75.62	77.29	78.33	78.96	79.58	79.38	.90 (79.58)	+0.21
8B	HMMT25	Mean	59.58	60.42	60.42	60.83	60.83	60.42	.75/.90 (60.83)	+0.42
		Tail	58.96	60.21	60.21	60.83	60.62	60.83	.75/1.00 (60.83)	+0.00
		Bottom10	58.54	60.42	59.79	60.21	60.83	60.62	.90 (60.83)	+0.21
8B	BRUMO25	Mean	78.54	80.42	80.63	81.25	81.04	81.25	.75/1.00 (81.25)	+0.00
		Tail	78.13	82.08	84.17	83.96	82.71	82.71	.50 (84.17)	+1.46
		Bottom10	78.75	79.17	80.21	80.21	80.21	80.63	1.00 (80.63)	+0.00

Table C.4: Top-η retention ablation for the TURN cross-temperature pool. Gain is measured against no filtering, i.e., $\eta = 1.00$.

Panel A. Overall aggregation comparison						
η	N	MajWeight avg.	MajScale avg.	Δ	W/T/L	Higher mean
0.10	54	62.28	59.52	-2.75	1/5/48	MajWeight
0.25	54	63.75	63.07	-0.67	8/23/23	MajWeight
0.50	54	63.40	63.83	+0.43	21/23/10	MajScale
0.75	54	62.63	63.78	+1.15	24/20/10	MajScale
0.90	54	62.61	63.65	+1.04	20/26/8	MajScale
1.00	54	62.36	63.25	+0.88	21/27/6	MajScale

Panel B. MajScale − MajWeight by setting and score source						
Stratum (N per η)	$\eta = 0.10$	$\eta = 0.25$	$\eta = 0.50$	$\eta = 0.75$	$\eta = 0.90$	$\eta = 1.00$
Base (27)	-2.79	-0.52	+0.69	+1.05	+1.00	+0.96
TURN (27)	-2.72	-0.83	+0.18	+1.26	+1.08	+0.80
Mean (18)	-3.30	-1.00	+0.29	+1.02	+0.86	+0.83
Tail (18)	-2.99	-1.05	+0.31	+1.18	+1.19	+1.15
Bottom10 (18)	-1.98	+0.03	+0.69	+1.26	+1.06	+0.67

Table C.5: MajWeight versus MajScale ablation. Panel A compares the average accuracy of direct score-weighted voting and min-max scaled voting across retention fractions. Panel B reports the mean difference, MajScale minus MajWeight, within setting and score-source strata; positive values favor MajScale. W/T/L denotes wins/ties/losses of MajScale against MajWeight.

Model	Dataset	Mix-Maj	RawConf	TNormConf	TN-MAD	TN-MAD-Full	Full − BestConf [95% CI]
1.5B	AIME25	38.75	38.75	38.75	38.75	38.75	0.00 [0.00, 0.00]
1.5B	HMMT25	17.92	17.92	18.33	18.33	18.33	0.00 [0.00, 0.00]
1.5B	BRUMO25	46.67	46.25	47.08	47.08	47.08	0.00 [0.00, 0.00]
4B	AIME25	86.67	86.67	86.67	86.67	86.67	0.00 [0.00, 0.00]
4B	HMMT25	62.08	62.50	67.50	66.67	69.17	+1.25 [-1.25, 4.17]
4B	BRUMO25	83.75	84.17	85.00	85.00	86.67	+1.25 [-0.42, 3.75]
8B	AIME25	79.17	77.92	78.75	79.17	78.33	-0.83 [-2.50, 0.00]
8B	HMMT25	60.00	60.00	60.00	60.42	60.83	+0.83 [0.00, 2.08]
8B	BRUMO25	81.25	81.25	82.92	82.50	82.92	0.00 [-1.25, 1.25]
Macro	-	61.81	61.71	62.78	62.73	63.19	+0.28 [-0.19, 0.74]

Table C.6: Same-pool scoring controls on identical mixed-temperature pools. `TN-MAD-Full` applies TN-MAD top-90% filtering plus scaled voting. The final column compares it with the better of `RawConf` and `TNormConf` on each problem.

Macro contrast	Difference [95% CI]
TN-MAD − Mix-Maj	+0.93 [0.19, 1.81]
TN-MAD − RawConf	+1.02 [0.32, 1.81]
TN-MAD − TNormConf	-0.05 [-0.32, 0.28]
TN-MAD-Full − Mix-Maj	+1.39 [0.37, 2.50]
TN-MAD-Full − BestConf	+0.28 [-0.19, 0.74]

Table C.7: Macro same-pool scoring contrasts in percentage points.

Single-temp ref.	A RawConf	B TN-MAD	C Merged RawConf	D Merged TN-MAD	D-C	AUROC RawConf	AUROC TN-MAD
$T = 0.4$	63.17	62.94	61.48	63.75	+2.27	0.59	0.80
$T = 0.6$	63.12	62.31	61.48	63.75	+2.27	0.59	0.80
$T = 0.8$	64.26	63.63	61.48	63.75	+2.27	0.59	0.80
$T = 1.0$	62.89	62.92	61.48	63.75	+2.27	0.59	0.80

Table C.8: Merged-pool factor ablation. `RawConf` denotes raw confidence. A and B use a single-temperature pool; C and D merge four temperature pools before top-25% retention. Accuracy values are macro averages; AUROC is measured in the merged pool.

D Additional Experiments and Diagnostics

This appendix reports trace-level diagnostics that support the main experiments without treating them as primary performance claims. Section D.1 visualizes TN-MAD score distributions for all model-dataset settings, and Section D.2 reports the token profile of retained traces under different scoring rules. Section D.3 provides a manually checked qualitative example of a valid reasoning path that appears only at a higher sampling temperature.

D.1 Score Distribution Diagnostics

Figures D.1–D.3 summarize TN-MAD score distributions for the regular cross-temperature pool $\{0.4, 0.6, 0.8, 1.0\}$. The panels separate correct and incorrect traces and use stacked colors to show the contribution of each sampling temperature. These diagnostics help assess whether high TN-MAD scores are associated with correct traces without concentrating the high-scoring region at a single source temperature.

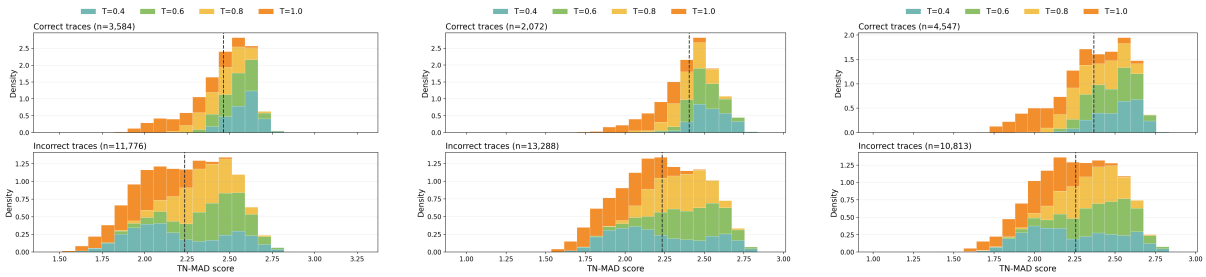
D.2 Retained-Trace Token Usage

Table D.1 compares the average token length of traces retained by no-rank selection, raw-confidence ranking, and TN-MAD ranking. No-rank selection retains traces without score-based ordering and serves as the control. This is a retained-set diagnostic only: the current evaluation still generates the full candidate pool before scoring, so the table does not claim total generation-token or wall-clock savings.

Selection	.10	.25	.50	.75	.90
No-rank	21471	21471	21471	21471	21471
RawConf	21398	26275	23532	22229	21615
	(-0.3%)	(+22.4%)	(+9.6%)	(+3.5%)	(+0.7%)
TN-MAD	**16672**	**17146**	**18005**	**19868**	**20661**
	(-22.3%)	**(-20.1%)**	**(-16.1%)**	**(-7.5%)**	**(-3.8%)**

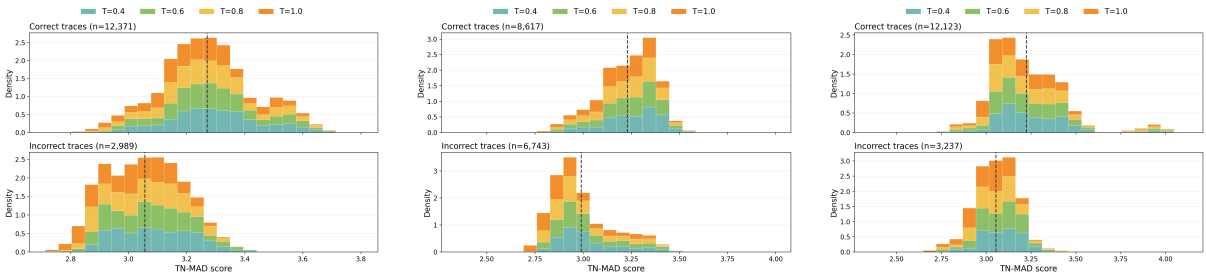
Table D.1: Average tokens per retained trace. Parentheses show relative changes against no-rank selection.

TN-MAD retains shorter traces than both raw confidence and the no-rank control across the evaluated retention values. Correctly reconstructing raw confidence reveals that it preferentially retains longer traces for $\eta \geq 0.25$ in this mixed-temperature setting, whereas TN-MAD retains shorter traces throughout. This remains a retained-set diagnostic: the current offline procedure does not establish online cost savings.



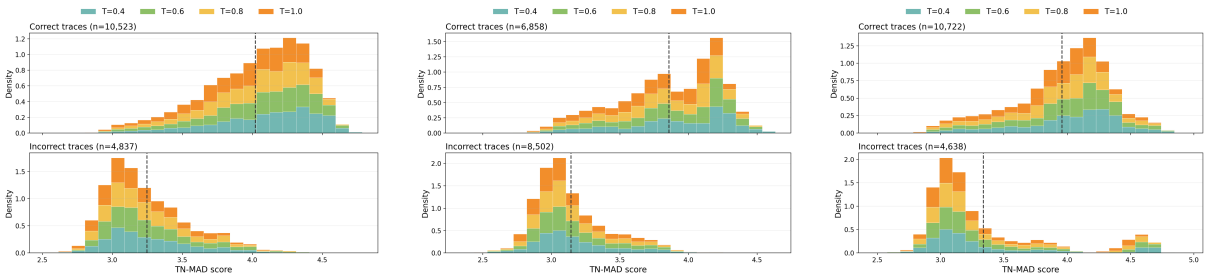
(a) AIME25 (b) HMMT25 (c) BRUMO25

Figure D.1: TN-MAD score distributions for DeepSeek-R1-Distill-Qwen-1.5B. Each panel separates correct and incorrect traces and colors bars by source temperature.



(a) AIME25 (b) HMMT25 (c) BRUMO25

Figure D.2: TN-MAD score distributions for Qwen3-4B-Thinking-2507. Each panel separates correct and incorrect traces and colors bars by source temperature.



(a) AIME25 (b) HMMT25 (c) BRUMO25

Figure D.3: TN-MAD score distributions for Qwen3-8B. Each panel separates correct and incorrect traces and colors bars by source temperature.

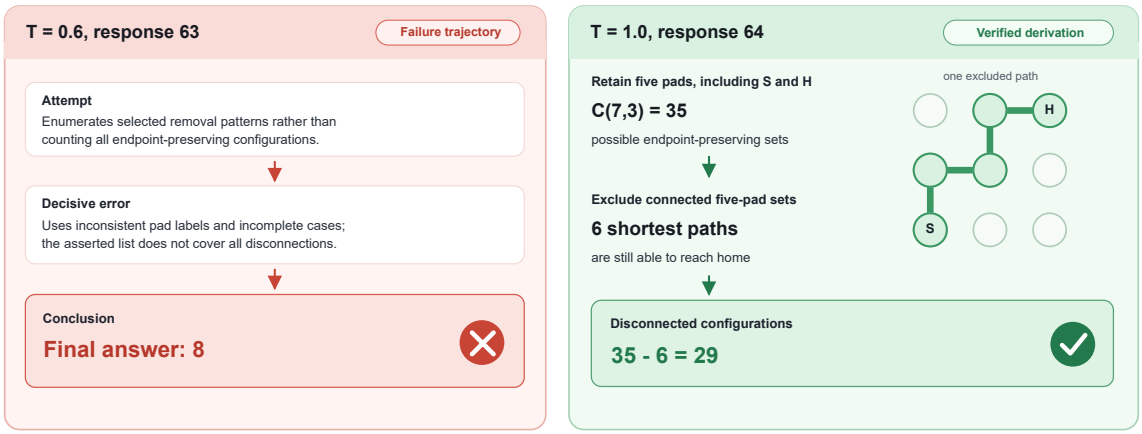


Figure D.4: A manually verified temperature-rescue case on HMMT25_12. All 128 traces at $T = 0.6$ and $T = 0.8$ miss the gold answer. At $T = 1.0$, response 64 supplies a valid counting argument: C_7^3 endpoint-preserving five-pad sets minus six connecting paths yields 29.

D.3 Case Study: A Valid Path Recovered at Higher Temperature

Figure D.4 illustrates a sampling-diversity case for HMMT25_12, which asks for the number of ways to remove four pads from a 3×3 grid while retaining opposite-corner endpoints but disconnecting them. Among 128 responses sampled at $T = 0.6$, none produces the gold answer 29; the same is true at $T = 0.8$. At $T = 1.0$, two responses produce 29. We manually inspect these answer-matched traces and use only response 64 as the positive trajectory, because it contains a valid derivation; the other answer-matched response does not.

The two displayed trajectories isolate what the increased sampling temperature makes available. The representative $T = 0.6$ response attempts an incomplete enumeration of removal sets and concludes with the incorrect answer 8. In contrast, the verified $T = 1.0$ response counts the configurations by their retained pads: after fixing the start and home pads, it chooses the remaining three retained pads in $C_7^3 = 35$ ways, excludes the six five-pad sets that form a connecting shortest path, and obtains $35 - 6 = 29$. Thus, this example shows that higher-temperature sampling can expose a sound solution path absent from the fixed-$T = 0.6$ samples.