SAGE: Answer-Conditioned Uncertainty Targets for Verbal Uncertainty Alignment

Kaiwen Shi[*] **Zheyuan Zhang**[*] **Yanfang Ye**[†]
University of Notre Dame
{kshi3, zzhang42, yye7}@nd.edu

Abstract

Large language models increasingly express uncertainty through natural-language statements, yet these expressions often fail to reflect the model's sampled behavior. We study verbal uncertainty alignment as a distributional calibration problem: the appropriate uncertainty target for a prompt should be estimated from repeated model outputs rather than from an isolated response. However, group rollouts alone are insufficient, since the resulting target must provide a useful training signal. Existing targets only partially satisfy this requirement. Therefore, we propose **SAGE** (**S**emantic-**A**nswer **G**uided **E**ntropy), a group-level uncertainty target that constructs an answer-conditioned uncertainty geometry over sampled responses. SAGE preserves categorical, numeric, and symbolic answer distinctions while maintaining a smooth and scale-preserving calibration signal. We further apply this target through **Group-Uncertainty Preference Optimization** (GUPO), an uncertainty-channel training framework that supervises verbal uncertainty expressions rather than the full response. Experiments across factual, mathematical, and multiple-choice reasoning tasks show improved uncertainty ranking, lower calibration error, and reduced overconfidence. Our code is available here.

1 Introduction

Recent progress in large language models (LLMs) (Ye et al., 2025; Chen et al., 2025b,a) has achieved remarkable success and expanded their use in open-ended generation tasks (Meng et al., 2024; Belkhouribchia and Pen, 2025). However, this success is accompanied by serious reliability challenges: model outputs can appear fluent and plausible even when the underlying answer is uncertain or unsupported (Farquhar et al., 2024). In real-world scenarios, hidden uncertainty may lead to hallucinations with severe consequences. These failures are not simply matters of imperfect text generation; they can mislead users, amplify unsafe decisions, and, in extreme cases, cause physical harm (Savage et al., 2024; Omar et al., 2025).

To mitigate such risks, researchers have explored different ways to represent model uncertainty. Internal signals, such as token probabilities (Gupta et al., 2024) or hidden states (Azaria and Mitchell, 2023), are informative but require model access and are difficult to interpret. Sampling-based signals reveal behavioral instability across repeated generations (Kuhn et al., 2023; Nikitin et al., 2024a), but they are costly. Meanwhile, asking an LLM to verbalize its own uncertainty provides a direct and practical signal (Xiong et al., 2023a; Lin et al., 2022; Zhou et al., 2025; Yang et al., 2026). A statement such as "I am uncertain" or "I am 80% confident" is immediately visible in the model output, available even in black-box settings, and easy for users or downstream applications to interpret. However, this convenience is useful only if the expressed uncertainty matches the reliability of the model's actual behavior, i.e., if the verbal uncertainty is calibrated (Kapoor et al., 2024). Without calibration, verbal uncertainty becomes unreliable and can turn model errors into misplaced trust.

A key challenge is constructing the target used to align verbal uncertainty. Since uncertainty should reflect the model's behavioral uncertainty rather than the surface form of a single answer, the central question is what target the model should learn (Geng et al., 2023). Existing targets capture different aspects of uncertainty, but each leaves an important gap. Maximum answer frequency (MAF) is answer-relevant but ignores semantic distance (Cole et al., 2023). Semantic Entropy (SE) handles paraphrases through meaning-equivalence clusters, but hard clustering makes the signal coarse and discontinuous (Kuhn et al., 2023). Kernel language entropy (KLE) is smoother, but generic embedding

[*]Equal contribution.
[†]Corresponding author.

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 25717–25737
October 24-29, 2026 ©2026 Association for Computational Linguistics

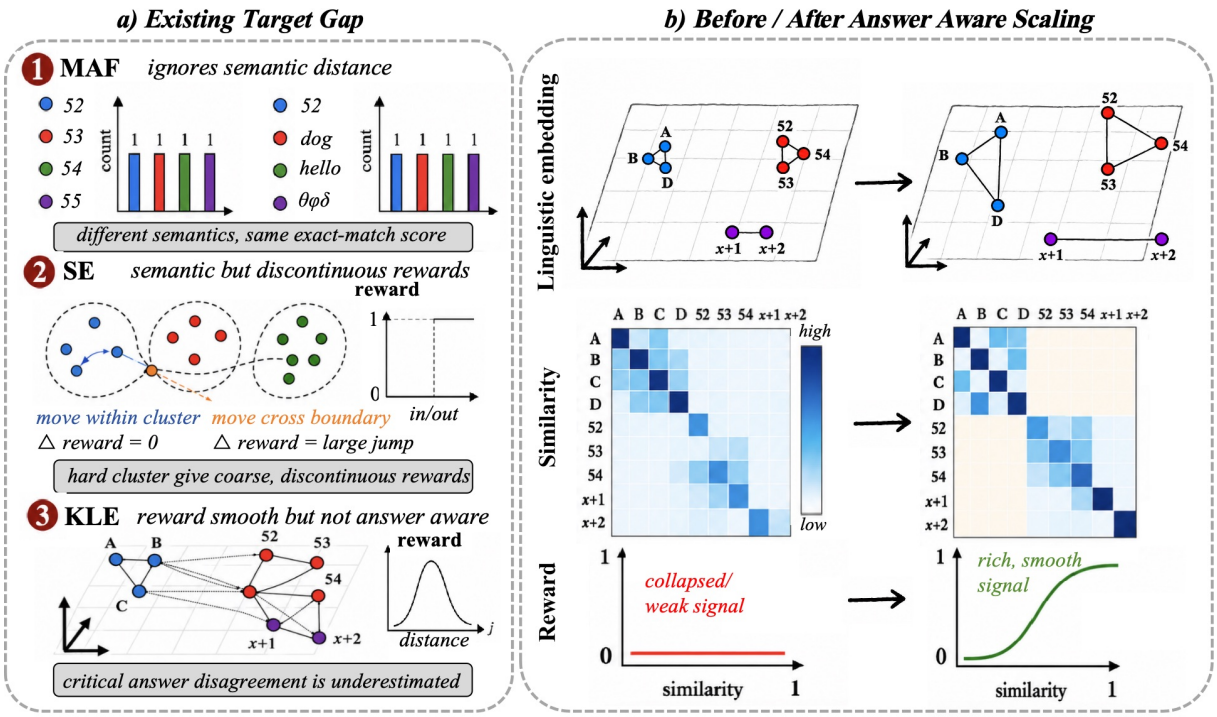

Figure 1: **Our motivation and key idea.** Existing group-level targets each miss a key requirement for verbal uncertainty calibration: MAF is answer-faithful but coarse, SE captures semantic equivalence but produces discontinuous rewards, and KLE is smooth but can underestimate task-critical answer disagreement in generic embedding space. SAGE addresses these limitations by injecting answer-aware structure into the response geometry while preserving smooth kernel-based entropy. This expands task-critical differences within compressed semantic neighborhoods and yields a smoother, more answer-faithful, and scale-preserving uncertainty target, providing a stronger reward signal for verbal uncertainty alignment.

geometry can blur task-critical answer distinctions, such as option labels, numeric values, or symbolic expressions (Nikitin et al., 2024a). Thus, target quality becomes an optimization bottleneck: **without a reliable uncertainty target, the reward signal can collapse into noise or shortcuts,** leaving the optimizer unable to distinguish calibrated uncertainty from superficial response patterns.

To address this target gap, we propose **SAGE** (**S**emantic-**A**nswer **G**uided **E**ntropy), a new group-level self-uncertainty target that reformulates verbal uncertainty alignment through answer-conditioned uncertainty geometry. The key idea is that an uncertainty target should not be determined solely by generic semantic variation; it should also reflect whether sampled generations remain compatible under the task-specific answer structure. This allows SAGE to preserve smooth reward variation while correcting a common failure mode of generic semantic metrics, where incompatible option labels, distinct numerical values, or non-equivalent symbolic expressions are treated as near-equivalent. By imposing this answer-conditioned structure, SAGE turns repeated generations into a structured uncertainty signal rather than an unorganized set of samples. Building on this target, we further introduce **Group-Uncertainty Preference Optimization** (GUPO), which uses SAGE as the supervision signal to align verbal uncertainty with sampled model behavior. Experiments on factual QA, mathematical reasoning, and multiple-choice understanding show that our framework improves uncertainty ranking, reduces calibration error, and mitigates overconfidence across distinct answer formats.

Our contributions are as follows:

- We show that verbal uncertainty alignment is a distributional calibration problem, and identify a reward-signal bottleneck that limits existing group-level uncertainty targets.
- We introduce **SAGE**, a semantic-answer guided entropy target, together with **GUPO**, an uncertainty-channel preference framework that optimizes verbal uncertainty expressions instead of full responses.

- We validate the approach across factual, mathematical, and multiple-choice reasoning tasks, where our framework achieves stronger overall performance than existing baselines across most evaluation metrics.

2 Problem formulation

2.1 Distributional Blindness

Existing work attempts to address uncertainty calibration through response-level methods, such as supervised fine-tuning (Jang et al., 2025) and pairwise preference optimization (Zhang et al., 2025b,a). However, a single response can reveal what the model sampled, but it cannot reveal whether the underlying probability mass is concentrated on that answer or dispersed across plausible alternatives, thus creating a structural mismatch for self-uncertainty calibration.

To be specific, supervised fine-tuning teaches the model to reproduce a target uncertainty expression by maximizing the likelihood of the supervised output tokens (Lin et al., 2022; Chaudhry et al., 2024; Jang et al., 2025), instead of directly learning how uncertainty should change with the dispersion of the model's own response distribution. Similarly, pairwise preference optimization establishes only a local gradient by updating the policy to maximize the log-likelihood margin between two static responses (Li et al., 2026; Zhang et al., 2025b), which lack sufficient information to determine whether the policy is sampling from a stable or highly entropic region. As a result, response-level training can adjust the form or scale of verbal uncertainty, but it does not naturally align uncertainty with the model's sampled response distribution.

2.2 Group-Uncertainty Preference Optimization

After constructing a group-level uncertainty target $t(G_x)$, we apply the calibration signal to the verbal uncertainty expression rather than to the full response. For a response $y_i = (z_i, v_i)$, the answer-bearing content z_i provides context, while the uncertainty expression v_i is the object of calibration. Let $\hat{u}(v_i) \in [0, 1]$ denote the uncertainty value expressed by v_i. We define

$$r_i = -\ell\big(\hat{u}(v_i), t(G_x)\big),$$

where larger values indicate that the expressed uncertainty is closer to the group-level uncertainty target.

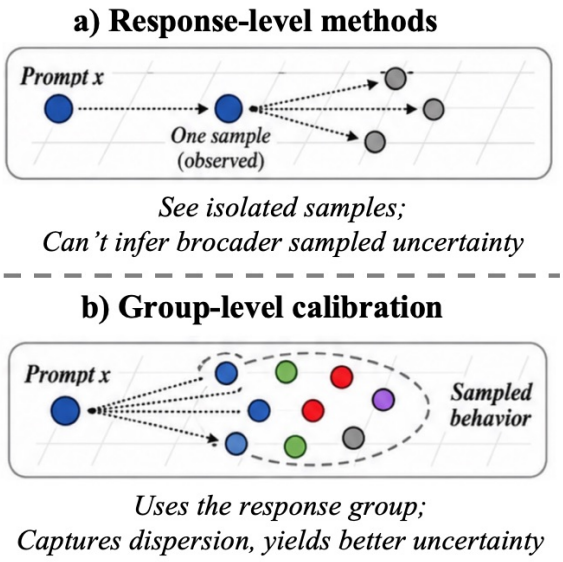

Figure 2: Illustration of the distributional gap in verbal uncertainty calibration. (a) A single response cannot reveal whether it comes from a stable or unstable response distribution. (b) Distributional calibration uses repeated samples: stable groups justify lower uncertainty, while dispersed groups require higher uncertainty.

We use these scores to form a group-relative preference distribution over uncertainty expressions:

$$p_i^{\text{cal}} = \frac{\exp(r_i/T)}{\sum_{j=1}^{K} \exp(r_j/T)},$$

where T controls preference sharpness. GUPO applies this preference to the uncertainty channel,

$$v_i \mid (x, z_i),$$

rather than treating the entire response as the optimization target. Conceptually, this corresponds to

$$\mathcal{L}_{\text{GUPO}} = -\mathbb{E}_{x,G_x}\left[\sum_{i=1}^{K} \text{stopgrad}(p_i^{\text{cal}}) \cdot \log \pi_\theta(v_i \mid x, z_i)\right].$$

In practice, this can be implemented by applying the calibration loss only to the uncertainty-expression span. The main focus of this work is the construction of $t(G_x)$, which we study next.

3 Methodology

3.1 Why Existing Group Targets Give Weak Rewards

Group rollouts expose the response distribution, but they do not automatically produce a useful uncertainty target. In group-based optimization, the

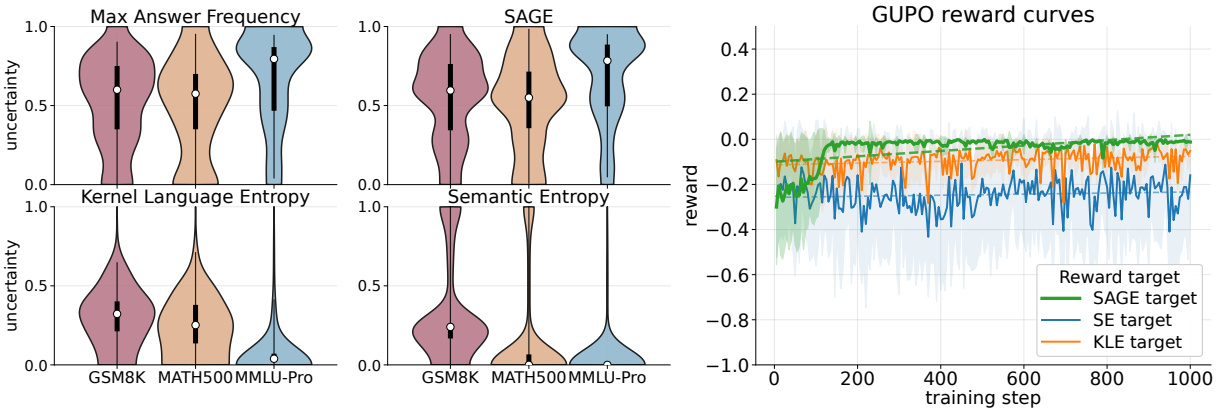

Figure 3: Uncertainty distributions produced by different group-level targets on the same 20 sampled responses from the evaluation set. Semantic Entropy (SE) and Kernel Language Entropy (KLE) can both underestimate answer-level uncertainty: SE collapses variation through hard semantic clusters, while KLE smooths over task-critical answer disagreement in embedding space. This motivates a target that is smooth, answer-aware, and scale-preserving.

model only receives a useful signal if the group-level target changes in ways that reflect meaningful uncertainty differences, while a poor target can make repeated sampling ineffective. If the target is too coarse, too discontinuous, or misaligned with answer-level disagreement, then the optimizer receives little guidance about how verbal uncertainty should change.

Weak Group Rewards. Figure 1 illustrates why existing group-level targets provide weak reward signals for verbal uncertainty calibration. Given a response group $G_x = \{y_i\}_{i=1}^K$ and extracted answers $\{a_i\}_{i=1}^K$, the simplest stability target is maximum answer frequency:

$$s_{\mathrm{MAF}}(G_x) = \max_a \frac{1}{K} \sum_{i=1}^{K} \mathbb{I}[a_i = a].$$

Since Maximum Answer Frequency (Cole et al., 2023) measures answer stability, its corresponding uncertainty target can be written as $t_{\mathrm{MAF}}(G_x) = 1 - s_{\mathrm{MAF}}(G_x)$. MAF is answer-faithful: if most samples produce the same final answer, the model should generally express lower uncertainty. However, as shown in Fig. 1(1), MAF only counts exact answer agreement. It maps semantically different response groups to the same score whenever their majority counts are identical:

$$G_x \neq G'_x, \qquad s_{\mathrm{MAF}}(G_x) = s_{\mathrm{MAF}}(G'_x).$$

Thus, MAF preserves final-answer agreement but discards semantic distance, reasoning variation, and secondary answer alternatives.

Semantic Entropy (SE) (Kuhn et al., 2023) improves on exact counting by clustering responses into meaning-equivalence classes. Let c_i denote the semantic cluster of response y_i and $p(c)$ be the empirical cluster frequency:

$$t_{\mathrm{SE}}(G_x) = -\sum_c p(c) \log p(c).$$

This handles paraphrases and aliases better than raw answer frequency. Yet, as shown in Fig. 1(2), SE depends on hard cluster membership. A small movement within the same cluster gives no reward change,

$$c_i = c'_i \quad \Rightarrow \quad \Delta t_{\mathrm{SE}} = 0,$$

while crossing a cluster boundary can cause a discrete jump. Therefore, SE produces coarse and discontinuous rewards: many meaningful changes in similarity or reasoning variation are invisible unless they alter the cluster assignment.

Kernel Language Entropy (KLE) (Nikitin et al., 2024a) replaces hard clusters with continuous similarities:

$$t_{\mathrm{KLE}}(G_x) = H_{\mathrm{vN}}(K), \qquad K_{ij} = k(y_i, y_j),$$

where K is a semantic similarity kernel over generated responses. This gives a smoother uncertainty signal than SE. However, as shown in Fig. 1(3), its smoothness comes from generic embedding geometry rather than the task-specific answer geometry. As a result, KLE may assign high similarity to outputs that are linguistically close but answer-wise incompatible:

$$k(y_i, y_j) \text{ high} \quad \not\Rightarrow \quad a_i \equiv a_j.$$

This is problematic for structured answer spaces such as multiple-choice labels, numerical answers, or symbolic expressions, where small surface differences can encode decisive answer disagreement.

Together, these limitations show that a useful uncertainty target must be both answer-sensitive and reward-smooth. Otherwise, the reward either collapses into coarse counts, discontinuous cluster changes, or misleading embedding similarity, leaving the optimizer with weak credit assignment for verbal uncertainty calibration.

Empirical Observation. Figure 3 reveals a reward-signal bottleneck. Under the same repeated-sampling groups, Maximum Answer Frequency preserves a broad stability range, whereas SE and KLE are compressed toward low uncertainty, thereby underestimating disagreement among sampled answers.

This compression arises because generic semantic similarity may miss task-specific answer disagreement. For example, options A, B, C, and D can be close in embedding space despite being mutually exclusive; nearby numeric answers such as 52, 53, and 54 may appear similar although only one is correct; and symbolic answers such as $x + 1$ and $x + 2$ can be linguistically close while mathematically distinct. Consequently, SE and KLE assign weakly separated uncertainty targets to groups that require different uncertainty levels. Their reward curves therefore improve only marginally and quickly plateau, suggesting limited guidance for adjusting verbal uncertainty. In contrast, SAGE preserves answer-level distinctions while maintaining a smooth signal, yielding a more informative reward trajectory.

3.2 SAGE: Semantic-Answer Guided Entropy

We propose **Semantic-Answer Guided Entropy** (SAGE) to construct such a target. SAGE keeps the continuous kernel view of linguistic entropy, but modifies the response geometry using task-specific answer equivalence.

For each response $y_i = (z_i, v_i)$, let $e(z_i)$ denote a continuous representation of the answer-bearing content. We define the linguistic kernel as

$$K_{\text{ling}}(y_i, y_j) = \exp\left(-\frac{\|e(z_i) - e(z_j)\|^2}{2\sigma_{\text{ling}}^2}\right),$$

which measures smooth semantic similarity between responses.

Let a_i be the extracted final answer from z_i. We first define a task-specific binary answer-equivalence kernel

$$K_{\text{ans}}^{\text{bin}}(a_i, a_j) \in \{0, 1\},$$

where equivalent answers receive value 1. Since a hard kernel may over-separate cross-answer pairs, we use a soft floor:

$$K_{\text{ans}}^{\text{soft}}(a_i, a_j) = c + (1-c)K_{\text{ans}}^{\text{bin}}(a_i, a_j), c \in [0, 1].$$

Here, $c = 0$ gives hard answer separation, while $c = 1$ removes answer geometry and recovers linguistic KLE.

The final answer-aware kernel is

$$K_{\text{A}}(y_i, y_j) = K_{\text{ling}}(y_i, y_j)^{\alpha} \left(K_{\text{ans}}^{\text{soft}}(a_i, a_j)\right)^{\beta},$$

where α and β control linguistic and answer-aware similarity. Diagonal entries are set to 1.

We normalize K_{A} into a density matrix:

$$P_{\text{A}} = \frac{K_{\text{A}}}{\text{tr}(K_{\text{A}})}.$$

SAGE is the normalized von Neumann entropy:

$$H_{\text{SAGE}}(G_x) = -\frac{1}{\log K} \text{tr}\left(P_{\text{A}} \log P_{\text{A}}\right).$$

Lower entropy indicates concentrated responses under linguistic and answer-aware similarity, while higher entropy indicates dispersion across meanings, reasoning paths, or final answers. We use this entropy as the group-level self-uncertainty target:

$$t(G_x) = H_{\text{SAGE}}(G_x).$$

3.3 Task-Specific Answer Equivalence

The answer-equivalence kernel adapts SAGE across answer spaces by preserving task-critical distinctions that generic embeddings may obscure.

Categorical answers. For multiple-choice tasks, option labels define equivalence:

$$K_{\text{ans}}^{\text{bin}}(a_i, a_j) = \mathbb{I}[a_i = a_j].$$

This prevents mutually exclusive options from being merged by embedding similarity.

Numeric answers. For numeric reasoning, final values are normalized before comparison:

$$K_{\text{ans}}^{\text{bin}}(a_i, a_j) = \mathbb{I}\big[\text{NormNum}(a_i) = \text{NormNum}(a_j)\big].$$

Benchmark	Method	Calibration			High-Confidence Accuracy		
		Brier ↓	ECE ↓	Spear. ↑	@80 ↑	@60 ↑	@50 ↑
MMLU-Pro	Direct Verbalized	0.126	0.246	0.069	0.303	0.312	0.325
	Direct Verbalized + DCA	0.113	0.222	0.124	0.356	0.412	0.445
	CoT	0.409	0.590	0.299	0.291	0.296	0.340
	CoT + DCA	0.094	0.179	0.100	0.319	0.342	0.355
	CSFT	0.045	0.120	0.195	0.347	0.388	0.380
	LoVeC-DPO	0.051	0.144	0.261	0.403	0.438	0.460
	GUPO + Semantic Entropy	0.214	0.316	0.090	0.188	0.183	0.180
	GUPO + Kernel Language Entropy	0.068	0.205	-0.010	0.163	0.167	0.180
	GUPO + SAGE	**0.037**	**0.113**	**0.572**	**0.412**	**0.512**	**0.570**
	Δ vs. Best Baseline	−0.008	−0.007	+0.273	+0.009	+0.074	+0.110
MATH-500	Direct Verbalized	0.359	0.546	0.246	0.563	0.617	0.660
	Direct Verbalized + DCA	0.388	0.578	0.171	0.550	0.617	0.580
	CoT	0.396	0.582	0.175	0.588	0.617	0.600
	CoT + DCA	0.287	0.479	0.074	0.588	0.567	0.580
	CSFT	0.064	0.143	**0.500**	**0.625**	0.700	**0.740**
	LoVeC-DPO	0.121	0.220	0.352	0.600	0.633	0.660
	GUPO + Semantic Entropy	0.190	0.282	0.426	0.575	0.650	0.720
	GUPO + Kernel Language Entropy	0.0205	0.0348	0.451	0.588	0.700	0.720
	GUPO + SAGE	**0.0202**	**0.0238**	0.463	0.613	**0.717**	**0.740**
	Δ vs. Best Baseline	−0.044	−0.119	−0.037	−0.012	+0.017	+0.000
TriviaQA	Direct Verbalized	0.394	0.551	0.132	0.388	0.433	0.440
	Direct Verbalized + DCA	0.235	0.402	0.507	0.450	0.517	0.570
	CoT	0.341	0.504	0.307	0.419	0.467	0.500
	CoT + DCA	0.209	0.368	0.523	0.469	0.508	0.560
	CSFT	0.119	0.177	0.241	0.494	0.517	0.540
	LoVeC-DPO	0.107	0.118	0.015	0.481	0.467	0.480
	GUPO + Semantic Entropy	0.153	0.247	0.404	0.480	0.530	0.540
	GUPO + Kernel Language Entropy	0.123	0.227	0.268	0.490	0.480	0.520
	GUPO + SAGE	**0.055**	**0.031**	**0.617**	**0.506**	**0.583**	**0.650**
	Δ vs. Best Baseline	−0.052	−0.087	+0.094	+0.012	+0.066	+0.080

Table 1: Calibration results across three benchmarks. Lower values are better for Brier score and ECE, while higher values are better for Spearman correlation and high-confidence subset accuracy. In our uncertainty framing, high-confidence subset accuracy corresponds to evaluating whether lower-uncertainty predictions are more reliable. Δ is computed against the best non-GUPO baseline for each metric.

This groups equivalent numeric forms while separating distinct values.

Symbolic answers. For symbolic mathematics, final expressions are normalized and checked for equivalence:

$$K_{\text{ans}}^{\text{bin}}(a_i, a_j) = \mathbb{I}[\text{Verify}(a_i, a_j) = 1].$$

This preserves mathematical equivalence beyond surface form.

Free-form answers. For open-ended factual QA, equivalence can be defined by normalized aliases or bidirectional entailment, grouping paraphrases while separating incompatible facts.

4 Experiments

4.1 Experimental Setup

Benchmarks. We evaluate on TriviaQA (Joshi et al., 2017), MATH-500 (Hendrycks et al., 2021), and MMLU-Pro (Wang et al., 2024). These benchmarks cover three representative answer structures: free-form factual answers, numeric or symbolic mathematical answers, and discrete multiple-choice options.

Compared methods. We compare GUPO with representative verbal uncertainty estimation and alignment baselines: Direct Verbalized prompting (Xiong et al., 2023a), CoT prompting (Cole et al., 2023), Direct Confidence Alignment (DCA)

(Zhang et al., 2025b), CSFT (Jang et al., 2025), and LoVeC-DPO (Zhang et al., 2025a). GUPO uses SAGE as the group-level self-uncertainty target. Confidence-based outputs are converted into uncertainty scores for comparison. Further details on the baselines are provided in Appendix B.3.

Metrics. We evaluate calibration quality and uncertainty ranking. Brier score and Expected Calibration Error (ECE) measure whether the model's expressed uncertainty is calibrated to empirical reliability. Spearman correlation evaluates whether the uncertainty scores correctly rank examples by reliability. We also report @80, @60, and @50, following the standard high-confidence subset accuracy protocol. In our uncertainty framing, these metrics evaluate whether predictions assigned lower uncertainty are more likely to be correct. Higher values are better for Spearman, @80, @60, and @50, while lower values are better for Brier score and ECE. Further details on the conversion between uncertainty and confidence-based metrics are provided in Appendix B.2.

4.2 Main Results

We evaluate verbal uncertainty using both calibration and decision-oriented metrics. Brier score and ECE measure whether uncertainty has the correct numerical scale, but our main focus is ranking and selective reliability. Spearman correlation measures whether lower uncertainty corresponds to higher correctness, while @80, @60, and @50 evaluate whether low-uncertainty predictions form more accurate subsets. These metrics directly test whether verbal uncertainty can guide downstream decisions such as trusting, verifying, or abstaining. This distinction is important because a model may appear numerically calibrated while still failing to identify which individual predictions are reliable. In practical deployments, uncertainty is useful only if it supports such instance-level decisions. Therefore, improvements in ranking and selective accuracy provide stronger evidence that uncertainty expressions are behaviorally meaningful.

Table 1 reports results on MMLU-Pro, MATH-500, and TriviaQA. Overall, **GUPO + SAGE** achieves the strongest profile. It obtains the best result on every metric for MMLU-Pro, the best Brier score and ECE on MATH-500 while remaining competitive on ranking metrics, and the clearest gains on TriviaQA across calibration, Spearman correlation, and high-confidence accuracy. These results show that GUPO + SAGE improves both the scale and, more importantly, the ordering of verbal uncertainty: predictions assigned lower uncertainty are more likely to be correct.

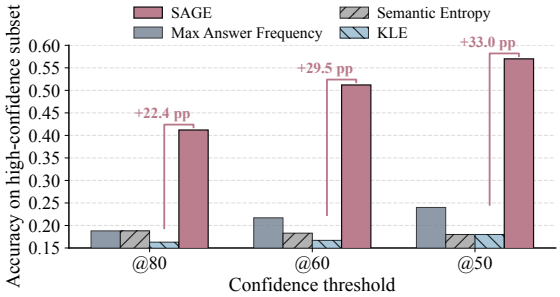

Figure 4: Ablation over group-level uncertainty targets on MMLU-Pro. SAGE achieves the best low-uncertainty subset accuracy across all thresholds, improving over the strongest baseline by $+22.4$, $+29.5$, and $+33.0$ percentage points. These results show that SAGE better identifies reliable low-uncertainty predictions.

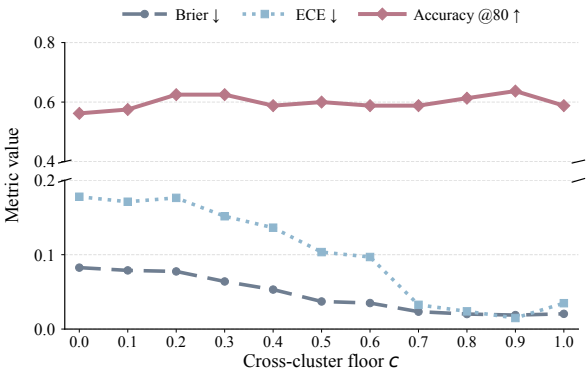

Figure 5: Ablation over the SAGE cross-cluster floor c on MATH-500. Increasing c improves calibration and low-uncertainty accuracy over the hard variant, while very large c moves the target closer to standard KLE. The results show that soft cross-cluster weighting provides a more effective training signal for SAGE.

4.3 Ablation Study

Uncertainty target. We first compare four targets for constructing the calibration preference distribution, while keeping the GUPO training framework fixed: **Max answer frequency**, which uses the most frequent extracted answer and converts answer stability into an uncertainty target; **Semantic Entropy**, which computes uncertainty over meaning clusters; **KLE**, which estimates entropy in a continuous semantic embedding space; and **SAGE**, which constructs an answer-conditioned uncertainty geometry for response groups. This ablation isolates the central question of our method:

Model	Setting	Brier ↓	ECE ↓	Spear. ↑	@80 ↑	@60 ↑	@50 ↑	Δ_μ ↓
Qwen2.5-1.5B-Instruct	Prompt	0.359	0.546	0.246	0.562	0.617	0.660	0.526
	GUPO+SAGE	**0.020**	**0.024**	**0.463**	**0.613**	**0.717**	**0.740**	**0.021**
	Δ vs. Prompt	−0.339	−0.522	+0.217	+0.051	+0.100	+0.080	−0.505
Qwen2.5-3B-Instruct	Prompt	0.401	0.619	-0.036	0.613	0.633	0.620	0.619
	GUPO+SAGE	**0.019**	**0.063**	**0.440**	**0.625**	**0.700**	**0.680**	**0.062**
	Δ vs. Prompt	−0.382	−0.556	+0.476	+0.012	+0.067	+0.060	−0.557
Qwen2.5-7B-Instruct	Prompt	0.389	0.602	0.381	**0.588**	0.583	0.660	0.602
	GUPO+SAGE	**0.024**	**0.035**	**0.485**	0.562	**0.650**	**0.700**	**0.035**
	Δ vs. Prompt	−0.365	−0.567	+0.104	−0.026	+0.067	+0.040	−0.567

Table 2: Calibration and uncertainty-alignment results across model sizes. Prompt denotes direct uncertainty elicitation without training, while **GUPO+SAGE** denotes uncertainty-channel training with the proposed semantic-answer guided group-level target. The mean-gap metric measures the absolute difference between the mean expressed uncertainty and the mean group-level uncertainty target. High-confidence subset accuracy is computed after converting expressed uncertainty into confidence; in our uncertainty framing, it evaluates whether lower-uncertainty predictions are more reliable. The delta rows report changes relative to Prompt; negative values are better for Brier score, ECE, and the mean-gap metric, while positive values are better for Spearman correlation and high-confidence subset accuracy.

whether a calibration target can simultaneously preserve answer-level disagreement, provide smooth reward variation, and avoid target-scale collapse. Frequency-based targets are answer-faithful but coarse; Semantic Entropy captures meaning equivalence but remains limited by hard cluster assignments; and KLE provides smoothness but can blur task-critical answer distinctions. In contrast, **SAGE is designed to satisfy all three requirements simultaneously**: answer faithfulness, reward smoothness, and scale preservation, making it a more effective target for converting group-level uncertainty into supervision for calibrated verbal uncertainty.

Effect of the cross-cluster floor. Figure 5 ablates the soft cross-cluster floor c in SAGE. When $c = 0$, the answer kernel becomes hard, preserving answer distinctions but potentially compressing the target scale. Increasing c retains some linguistic similarity across different answer clusters, producing a smoother and more usable training signal. Very large c, however, moves the target toward standard KLE and weakens the answer-aware correction. Overall, SAGE benefits from soft, scale-preserving answer separation rather than hard clustering or purely linguistic similarity.

Transferability Across Model Scales. Table 2 evaluates GUPO across Qwen2.5-1.5B-Instruct, Qwen2.5-3B-Instruct, and Qwen2.5-7B-Instruct. Compared with direct prompting, GUPO+SAGE consistently reduces Brier score and ECE, and generally improves Spearman correlation. These results show that the proposed uncertainty-alignment framework transfers across model scales rather than relying on a single backbone.

Detailed ablation results and additional analyses are reported in Appendix C.

5 Conclusion

We study verbal uncertainty alignment through behavioral uncertainty. Instead of treating uncertainty as a property of a single response, we define it by the stability of sampled outputs under the same prompt, framing calibration as a distributional problem. To derive a group-level target, we introduce SAGE, which combines continuous linguistic similarity with soft answer-equivalence structure. SAGE yields a smooth, answer-aware, scale-preserving uncertainty signal across factual, mathematical, and multiple-choice tasks. We further propose GUPO, which applies this signal to the verbal uncertainty channel rather than optimizing the full response. Experiments on TriviaQA, MATH-500, and MMLU-Pro show that GUPO improves alignment between expressed uncertainty and sampled model behavior, reducing overconfidence across task formats. These results demonstrate that group-level behavioral uncertainty can be distilled into single-response uncertainty expressions when both the target and optimization channel are properly designed.

Acknowledgments

This work was partially supported by the NSF under grants IIS-2533550, IIS-2321504, IIS-2217239, CNS-2426514, and CMMI-2146076, Notre Dame Strategic Framework Research Grant (2025), and Notre Dame Poverty Research Package (2025). Any expressed opinions, findings, and conclusions or recommendations are those of the authors and do not necessarily reflect the views of the sponsors.

Limitations

This work aligns verbal uncertainty with behavioral self-uncertainty rather than external correctness. The two are related but not identical: a model may be stable but wrong, or unstable while occasionally correct. Therefore, our method should be viewed as complementary to correctness-based calibration, retrieval-based verification, and external validation rather than as a replacement for them.

SAGE relies on answer extraction and task-specific equivalence rules to estimate group-level uncertainty. We use different normalization or equivalence procedures for factual, numeric, symbolic, and multiple-choice answers, but these procedures may still introduce noise, especially for open-ended tasks with ambiguous answer boundaries. In practice, this means that the quality of the uncertainty target depends partly on the reliability of the answer-equivalence module.

Finally, applying calibrated verbal uncertainty in high-stakes domains requires particular care. Better uncertainty expressions can help users recognize potentially unreliable outputs, but model predictions may still be affected by factual errors, incomplete evidence, hidden biases, or failures in the underlying reasoning process. A low-uncertainty response may still be incorrect, especially when the task involves domain-specific evidence or constraints. This is especially relevant in areas such as medicine, food safety, finance, and legal decision-making, where errors may have substantial real-world consequences. In these settings, model uncertainty can serve as a useful reference for risk awareness and verification, but final decisions should still be made by human experts and grounded in domain evidence, external validation, and appropriate institutional safeguards.

References

Amos Azaria and Tom M. Mitchell. 2023. The internal state of an llm knows when its lying. *ArXiv*, abs/2304.13734.

Han Bao, Zheyuan Zhang, Pengcheng Jing, Zhengqing Yuan, Kaiwen Shi, and Yanfang Ye. 2026. Driftbench: Diagnosing cooperative breakdowns in llm agents under input faults via multi-turn interaction. *arXiv preprint arXiv:2602.02455*.

Jamal Belkhouribchia and Joeri J. Pen. 2025. Large language models in clinical nutrition: an overview of its applications, capabilities, limitations, and potential future prospects. *Frontiers in Nutrition*, 12.

Arslan Chaudhry, Sridhar Thiagarajan, and Dilan Görür. 2024. Finetuning language models to emit linguistic expressions of uncertainty. *ArXiv*, abs/2409.12180.

Chaoran Chen, Zhiping Zhang, Bingcan Guo, Shang Ma, Ibrahim Khalilov, Simret Gebreegziabher, Yanfang Ye, Ziang Xiao, Yaxing Yao, Tianshi Li, and 1 others. 2025a. The obvious invisible threat: Llm-powered gui agents' vulnerability to fine-print injections. In *Soups*. The Twenty-First Symposium on Usable Privacy and Security (SOUPS).

Chaoran Chen, Daodao Zhou, Yanfang Ye, Toby Jia-jun Li, and Yaxing Yao. 2025b. Clear: Towards contextual llm-empowered privacy policy analysis and risk generation for large language model applications. In *Proceedings of the 30th International Conference on Intelligent User Interfaces*, pages 277–297.

Jeremy R. Cole, Michael J.Q. Zhang, Daniel Gillick, Julian Martin Eisenschlos, Bhuwan Dhingra, and Jacob Eisenstein. 2023. Selectively answering ambiguous questions. *ArXiv*, abs/2305.14613.

Ekaterina Fadeeva, Aleksandr Rubashevskii, Artem Shelmanov, Sergey Petrakov, Haonan Li, Hamdy Mubarak, Evgenii Tsymbalov, Gleb Kuzmin, Alexander Panchenko, Timothy Baldwin, and 1 others. 2024. Fact-checking the output of large language models via token-level uncertainty quantification. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 9367–9385.

Ekaterina Fadeeva, Roman Vashurin, Akim Tsvigun, Artem Vazhentsev, Sergey Petrakov, Kirill Fedyanin, Daniil Vasilev, Elizaveta Goncharova, Alexander Panchenko, Maxim Panov, and 1 others. 2023. Lm-polygraph: Uncertainty estimation for language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 446–461.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. 2024. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630:625 – 630.

Jiahui Geng, Fengyu Cai, Yuxia Wang, Heinz Koeppl, Preslav Nakov, and Iryna Gurevych. 2023. A survey

of confidence estimation and calibration in large language models. In *North American Chapter of the Association for Computational Linguistics*.

Neha Gupta, Harikrishna Narasimhan, Wittawat Jitkrittum, Ankit Singh Rawat, Aditya Krishna Menon, and Sanjiv Kumar. 2024. Language model cascades: Token-level uncertainty and beyond. *ArXiv*, abs/2404.10136.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Xiaodong Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *ArXiv*, abs/2103.03874.

Yuelin Hu, Zhengxue Cheng, Wei Liu, and Li Song. 2026. Entropy-gated selective policy optimization: Token-level gradient allocation for hybrid training of large language models. *arXiv preprint arXiv:2602.03309*.

Jiatan Huang, Zheyuan Zhang, Tianyi Ma, Mingchen Li, Yaning Zheng, Yanfang Ye, and Chuxu Zhang. 2026a. Glen-bench: A graph-language based benchmark for nutritional health. *arXiv preprint arXiv:2601.18106*.

Jiatan Huang, Zheyuan Zhang, Kaiwen Shi, Yanfang Ye, and Chuxu Zhang. 2026b. Evolverouter: Co-evolving routing and prompt for multi-agent question answering. *arXiv preprint arXiv:2604.05149*.

Chaeyun Jang, Moonseok Choi, Yegon Kim, Hyungi Lee, and Juho Lee. 2025. Verbalized confidence triggers self-verification: Emergent behavior without explicit reasoning supervision. *ArXiv*, abs/2506.03723.

Mingjian Jiang, Yangjun Ruan, Prasanna Sattigeri, Salim Roukos, and Tatsunori B. Hashimoto. 2024. Graph-based uncertainty metrics for long-form language model outputs. *ArXiv*, abs/2410.20783.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *ArXiv*, abs/1705.03551.

Mingxuan Ju, Tong Zhao, Wenhao Yu, Neil Shah, and Yanfang Ye. 2023. Graphpatcher: mitigating degree bias for graph neural networks via test-time augmentation. *Advances in neural information processing systems*, 36:55785–55801.

Sanyam Kapoor, Nate Gruver, Manley Roberts, Katherine M. Collins, Arka Pal, Umang Bhatt, Adrian Weller, Samuel Dooley, Micah Goldblum, and Andrew Gordon Wilson. 2024. Large language models must be taught to know what they don't know. *ArXiv*, abs/2406.08391.

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

Chen Li, Xiaolin Hu, Songzhu Zheng, Jiawei Zhou, and Chaoran Chen. 2026. Orce: Order-aware alignment of verbalized confidence in large language models.

Peiyu Li, Xiaobao Huang, Ting Hua, and Nitesh V Chawla. 2025a. Crochetbench: Can vision-language models move from describing to doing in crochet domain? *arXiv*.

Peiyu Li, Xiaobao Huang, Yijun Tian, and Nitesh V Chawla. 2024. Cheffusion: Multimodal foundation model integrating recipe and food image generation. In *CIKM*.

Peiyu Li, Xiuxiu Tang, Si Chen, Ying Cheng, Ronald Metoyer, Ting Hua, and Nitesh V Chawla. 2025b. Adaptive testing for llm evaluation: A psychometric alternative to static benchmarks. *arXiv*.

Stephanie C. Lin, Jacob Hilton, and Owain Evans. 2022. Teaching models to express their uncertainty in words. *Trans. Mach. Learn. Res.*, 2022.

Tianyi Ma, Yiyue Qian, Zheyuan Zhang, Zehong Wang, Xiaoye Qian, Feifan Bai, Yifan Ding, Xuwei Luo, Shinan Zhang, Keerthiram Murugesan, and 1 others. 2026. Autodata: A multi-agent system for open web data collection. *Advances in Neural Information Processing Systems*, 38:173416–173448.

Potsawee Manakul, Adian Liusie, and Mark John Francis Gales. 2023. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *ArXiv*, abs/2303.08896.

Xiangbin Meng, Xiangyu Yan, Kuo Zhang, Da Liu, Xiaosheng Cui, Yaodong Yang, Muhan Zhang, Chunxia Cao, Jingjia Wang, Xuliang Wang, Jun Gao, Yuangengshuo Wang, Jiaming Ji, Zifeng Qiu, Muzi Li, Cheng Qian, Tianze Guo, Shuang Ma, Zeying Wang, and 6 others. 2024. The application of large language models in medicine: A scoping review. *iScience*, 27.

Alexander Nikitin, Jannik Kossen, Yarin Gal, and Pekka Marttinen. 2024a. Kernel language entropy: Fine-grained uncertainty quantification for llms from semantic similarities. *ArXiv*, abs/2405.20003.

Alexander Nikitin, Jannik Kossen, Yarin Gal, and Pekka Marttinen. 2024b. Kernel language entropy: Fine-grained uncertainty quantification for llms from semantic similarities. *Advances in Neural Information Processing Systems*, 37:8901–8929.

Mahmud Omar, Vera Sorin, Jeremy D. Collins, David L. Reich, Robert Freeman, Nicholas Gavin, Alexander W. Charney, Lisa Stump, Nicola Luigi Bragazzi, Girish N. Nadkarni, and Eyal Klang. 2025. Multi-model assurance analysis showing large language models are highly vulnerable to adversarial hallucination attacks during clinical decision support. *Communications Medicine*, 5.

Thomas Savage, John Wang, Robert J. Gallo, Abdessalem Boukil, Vishwesh Patel, Seyed Amir Ahmad Safavi-Naini, Ali Soroush, and Jonathan H Chen.

2024. Large language model uncertainty proxies: discrimination and calibration for medical diagnosis and treatment. *Journal of the American Medical Informatics Association : JAMIA*.

Kaiwen Shi, Weixiang Sun, Zheyuan Zhang, Lichao Sun, Nitesh V Chawla, and Yanfang Ye. 2026a. Citeaudit: You cited it, but did you read it? a benchmark for verifying scientific references in the llm era. *arXiv preprint arXiv:2602.23452*.

Kaiwen Shi, Zheyuan Zhang, Han Bao, Colby Nelson, and Yanfang Ye. 2026b. Confidence laundering in agent systems: Why uncertainty needs a latent carrier. *arXiv preprint arXiv:2606.20662*.

Kaiwen Shi, Zheyuan Zhang, Zhengqing Yuan, Keerthiram Murugesan, Vincent Galassi, Chuxu Zhang, and Yanfang Ye. 2026c. Ng-router: Graph-supervised multi-agent collaboration for nutrition question answering. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7508–7527.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max W.F. Ku, Kai Wang, Alex Zhuang, Rongqi "Richard" Fan, Xiang Yue, and Wenhu Chen. 2024. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *ArXiv*, abs/2406.01574.

Zehong Wang, Zheyuan Liu, Tianyi Ma, Jiazheng Li, Zheyuan Zhang, Xingbo Fu, Yiyang Li, Zhengqing Yuan, Wei Song, Yijun Ma, and 1 others. 2025. Graph foundation models: A comprehensive survey. *arXiv preprint arXiv:2505.15116*.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2023a. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. *ArXiv*, abs/2306.13063.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2023b. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. *arXiv preprint arXiv:2306.13063*.

Siheng Xiong, Oguzhan Gungordu, Blair Johnson, James C Kerce, and Faramarz Fekri. 2026. Scaling search-augmented llm reasoning via adaptive information control. *arXiv preprint arXiv:2602.01672*.

Siheng Xiong, Ali Payani, and Faramarz Fekri. 2025a. Enhancing language model reasoning with structured multi-level modeling. In *The Fourteenth International Conference on Learning Representations*.

Siheng Xiong, Ali Payani, Yu'an Yang, and Faramarz Fekri. 2025b. Deliberate reasoning in language models as structure-aware planning with an accurate world model. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 31900–31931.

Qwen An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxin Yang, Jingren Zhou, Junyang Lin, and 25 others. 2024. Qwen2.5 technical report. *ArXiv*, abs/2412.15115.

Zhichao Yang, Caiqi Zhang, Ruihan Yang, Chengzu Li, Nigel Collier, and Deqing Yang. 2026. When calibration rankings reverse: Accuracy-controlled evaluation for fair comparison of llms. *arXiv preprint arXiv:2606.30814*.

Yanfang Ye, Tao Li, Qingshan Jiang, Zhixue Han, and Li Wan. 2009. Intelligent file scoring system for malware detection from the gray list. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1385–1394.

Yanfang Ye, Zheyuan Zhang, Tianyi Ma, Zehong Wang, Yiyang Li, Shifu Hou, Weixiang Sun, Kaiwen Shi, Yijun Ma, Wei Song, and 1 others. 2025. Llms4all: A review of large language models across academic disciplines. *arXiv preprint arXiv:2509.19580*.

Caiqi Zhang, Fangyu Liu, Marco Basaldella, and Nigel Collier. 2024. Luq: Long-text uncertainty quantification for llms. *ArXiv*, abs/2403.20279.

Caiqi Zhang, Xiaochen Zhu, Chengzu Li, Nigel Collier, and Andreas Vlachos. 2025a. Reinforcement learning for better verbalized confidence in long-form generation. *ArXiv*, abs/2505.23912.

Glenn Zhang, Treasure Mayowa, Jason Fan, Yicheng Fu, Aaron Sandoval, Sean O'Brien, and Kevin Zhu. 2025b. Direct confidence alignment: Aligning verbalized confidence with internal confidence in large language models. *ArXiv*, abs/2512.11998.

Zheyuan Zhang, Lin Ge, Hongjiang Li, Weicheng Zhu, Chuxu Zhang, and Yanfang Ye. 2026a. Mapro: Recasting multi-agent prompt optimization as maximum a posteriori inference. In *Findings of the Association for Computational Linguistics: EACL 2026*, pages 4458–4480.

Zheyuan Zhang, Kaiwen Shi, Han Bao, Zehong Wang, Tianyi Ma, and Yanfang Ye. 2026b. Why semantic entropy fails: Geometry-aware and calibrated uncertainty for policy optimization. *arXiv preprint arXiv:2605.21801*.

Zheyuan Zhang, Kaiwen Shi, Zhengqing Yuan, Zehong Wang, Tianyi Ma, Keerthiram Murugesan, Vincent Galassi, Chuxu Zhang, and Yanfang Ye. 2025c. Agentrouter: A knowledge-graph-guided llm router for collaborative multi-agent question answering. *arXiv preprint arXiv:2510.05445*.

Zheyuan Zhang, Zehong Wang, Tianyi Ma, Varun Sameer Taneja, Sofia Nelson, Nhi Ha Lan Le, Keerthiram Murugesan, Mingxuan Ju, Nitesh V Chawla, Chuxu Zhang, and 1 others. 2025d. Mopihfrs: A multi-objective personalized health-aware food recommendation system with llm-enhanced interpretation. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1*, pages 2860–2871.

Ej Zhou, Caiqi Zhang, Tiancheng Hu, Chengzu Li, Nigel Collier, Ivan Vulić, and Anna Korhonen. 2025. Beyond the final layer: Intermediate representations for better multilingual calibration in large language models. *arXiv preprint arXiv:2510.03136*.

A Related Work

LLMs (Ye et al., 2025; Xiong et al., 2025b,a, 2026; Li et al., 2024, 2025b,a; Ma et al., 2026) and graph learning (Wang et al., 2025; Zhang et al., 2025d; Ju et al., 2023; Ye et al., 2009) have advanced rapidly, providing complementary capabilities for language-centric reasoning and modeling complex relational structures, respectively. Building on this progress, LLM-driven agents have gained prominence for their ability to plan, interact, and solve complex tasks with limited human oversight (Zhang et al., 2026a, 2025c; Shi et al., 2026c; Huang et al., 2026b,a; Bao et al., 2026). A central practical challenge accompanying these successes is hallucination: models confidently producing incorrect or fabricated facts (Shi et al., 2026b,a). This leads to the growing research of reliably quantifying and leveraging model uncertainty to faithfully represents model capabilities to recognize and mitigate hallucinations in trustworthy AI.

A primary line of work estimates uncertainty directly from model predictions, where token-level entropy is used as a proxy for confidence. These methods interpret uncertainty as dispersion over next-token distributions and have been widely used in early studies (Fadeeva et al., 2023, 2024). More recent approaches incorporate such signals into training, for example by using entropy to modulate gradient magnitude while preserving update direction (Hu et al., 2026). While effective for capturing local prediction uncertainty, these approaches remain fundamentally limited: token-level entropy reflects lexical ambiguity, but does not capture higher-level semantic structure across complete responses. As a result, it is often insufficient for reasoning-intensive tasks where correctness depends on global coherence rather than local token uncertainty.

To address this limitation, a growing body of work shifts to *response-level uncertainty*, estimating confidence from agreement across multiple generated samples. Early approaches quantify agreement through exact-match frequency or consistency scores (Xiong et al., 2023b), while later work extends this idea to semantic and factual consistency using external evaluators or LLM-based judgments (Manakul et al., 2023; Zhang et al., 2024; Jiang et al., 2024). Entropy-based formulations provide a more principled probabilistic interpretation by modeling distributions over semantic outcomes, including clustering-based semantic entropy (Kuhn et al., 2023), its discrete variant (Farquhar et al., 2024), and similarity-based extensions such as kernel entropy (Nikitin et al., 2024b) and methods beyond entropy (Zhang et al., 2026b).

B Training Details

Overview. All models are trained in two stages. First, we perform format supervised fine-tuning to make the model reliably produce task-specific answer and uncertainty formats. Second, starting from the format-SFT checkpoint, we train verbal uncertainty using group-level uncertainty targets. This second stage includes supervised calibration baselines and our uncertainty-channel preference training. All target variants share the same sampled response groups, data split, and evaluation protocol.

Hardware and software

The main experiments use `Qwen/Qwen2.5-1.5B-Instruct` and are run on the PSC Bridges-2 cluster using the GPU-shared partition with a single NVIDIA V100-32GB GPU. For model-scale transfer experiments, `Qwen2.5-3B-Instruct` and `Qwen2.5-7B-Instruct` are trained on NVIDIA H100 GPUs. Training is implemented with TRL and Hugging Face `transformers`. Offline sampling and online rollouts use vLLM. Training and vLLM inference use fp16. For NLI-based semantic equivalence components, we use fp32 when required for attention compatibility. All experiments use seed 42 unless otherwise specified.

B.1 Training Model Selection

We use **Qwen2.5-1.5B-Instruct** (Yang et al., 2024) as the primary training model. This choice is motivated by both methodological and practical considerations. Methodologically, verbal uncertainty calibration should be evaluated on a model that still exhibits visible uncertainty and non-trivial instability under repeated sampling; overly strong models may saturate on some benchmarks and make calibration improvements harder to diagnose. Practically, the 1.5B model enables efficient group rollout collection and preference optimization, which is important because GUPO requires multiple sampled responses for each prompt to estimate the SAGE target. Therefore, Qwen2.5-1.5B-Instruct provides a suitable testbed for studying whether SAGE can produce informative group-level supervision for uncertainty alignment.

To verify that the observed gains are not specific to a single model scale, we further evaluate the same training framework on **Qwen2.5-3B-Instruct** and **Qwen2.5-7B-Instruct**. As shown in Table 2, GUPO+SAGE consistently improves calibration across model sizes. In particular, it substantially reduces Brier score, ECE, and the mean uncertainty-target gap Δ_μ, while improving Spearman correlation and high-confidence subset accuracy in most settings. These results suggest that the benefit of SAGE is not tied to the 1.5B model alone, but transfers across different model capacities.

B.2 Metric Details

Our model produces verbal uncertainty scores, while some standard calibration metrics are conventionally defined over confidence scores. For compatibility with prior work, we convert expressed uncertainty into the corresponding confidence score before computing Brier score, ECE, and threshold-based accuracy. Thus, high-confidence subset accuracy also has a direct uncertainty interpretation: examples above a confidence threshold are equivalently examples below the corresponding uncertainty threshold.

Specifically, @80, @60, and @50 report the accuracy of examples whose converted confidence score exceeds each threshold. In the uncertainty view, these metrics measure whether examples assigned lower uncertainty are more reliable. Therefore, higher @80, @60, and @50 indicate that the model's low-uncertainty predictions are more likely to be correct.

B.3 Baseline Details

We compare GUPO with representative baselines that cover prompting-based uncertainty elicitation, post-hoc calibration, supervised calibration training, and preference-based verbal uncertainty alignment.

Direct Verbalized. Direct Verbalized prompting asks the model to produce an answer together with an explicit verbal uncertainty statement. This baseline evaluates the model's native ability to express uncertainty without additional calibration training.

Chain-of-Thought Prompting. The CoT baseline asks the model to generate intermediate reasoning before producing the final answer and uncertainty expression. This tests whether adding reasoning improves the reliability of verbal uncertainty by making the answer process more explicit.

Direct Confidence Alignment. Direct Confidence Alignment (DCA) is a post-hoc calibration baseline that adjusts verbal confidence scores after generation. Since DCA operates on confidence values, we convert its calibrated confidence scores into uncertainty scores when comparing uncertainty alignment.

CSFT. CSFT uses supervised fine-tuning to train the model to produce calibrated verbal uncertainty expressions. Unlike GUPO, which uses group-level uncertainty targets and applies supervision to the uncertainty channel, CSFT follows a supervised response-level training objective.

LoVeC-DPO. LoVeC-DPO is a preference-based verbal calibration baseline. It improves verbal confidence expression through preference optimization, and we convert its confidence outputs into uncertainty scores for comparison. We include it to compare GUPO with an existing preference-based alignment approach.

GUPO with Alternative Targets. In addition to external baselines, we also evaluate GUPO with different group-level targets, including Semantic Entropy and Kernel Language Entropy. These variants use the same GUPO training framework as SAGE, but differ in how the group-level uncertainty target is constructed. This comparison isolates the contribution of SAGE from the optimization framework itself.

Dataset splits. We evaluate on three benchmarks with different answer structures. MATH-500 tests

mathematical reasoning with numeric or symbolic final answers. TriviaQA tests factual question answering with short free-form textual answers. MMLU-Pro tests multiple-choice reasoning with discrete option labels. We use fixed train and validation splits for all calibration experiments. MATH-500 contains 500 prompts, split into 400 training prompts and 100 validation prompts. TriviaQA uses 2,000 prompts, split into 1,600 training prompts and 400 validation prompts. MMLU-Pro uses 4,000 prompts, split into 3,200 training prompts and 800 validation prompts. The same split is used across target variants and training methods.

Offline response sampling for target construction

For each prompt x, we sample a group

$$G_x = \{y_1, \ldots, y_K\}, \qquad y_i \sim \pi_{\theta_0}(\cdot \mid x),$$

where θ_0 is the base model and $K = 20$. All offline sampling uses vLLM with temperature 0.7 and top-$p = 0.9$. For MATH-500, the model produces chain-of-thought reasoning and a final `\boxed{·}` answer, with `max_new_tokens` = 512 and `max_model_len` = 1536. For TriviaQA, the model produces a short answer in an explicit `Answer:/Uncertainty:` format, with `max_new_tokens` = 64 and `max_model_len` = 512. For MMLU-Pro, the model produces chain-of-thought reasoning and ends with `Answer: <letter>`, with `max_new_tokens` = 320 and `max_model_len` = 2560.

Answer extraction and equivalence. For each sampled response y_i, we extract the answer-bearing content z_i, the final answer a_i, and, when present, the verbal uncertainty expression v_i. For MATH-500, we extract the final boxed answer and apply numeric or symbolic equivalence checking when possible. For TriviaQA, we normalize short answers by lowercasing, removing punctuation and articles, and using alias or semantic equivalence when available. For MMLU-Pro, we extract the final option label. These extracted answers are used for maximum answer frequency, semantic clustering, and the answer-equivalence kernel in SAGE.

Group-level target construction. Each target variant maps the response group G_x to a scalar uncertainty target

$$t(G_x) \in [0, 1],$$

where larger values indicate greater uncertainty in the sampled response group. Maximum answer frequency is first computed as

$$s_{\mathrm{MAF}}(G_x) = \max_a \frac{1}{K} \sum_{i=1}^{K} \mathbb{I}[a_i = a].$$

Since maximum answer frequency measures answer stability, its uncertainty version is given by

$$t_{\mathrm{MAF}}(G_x) = 1 - s_{\mathrm{MAF}}(G_x).$$

Semantic Entropy clusters responses into meaning-equivalence classes and computes uncertainty over the empirical cluster distribution. KLE estimates response dispersion in a continuous embedding space. SAGE combines linguistic similarity with answer-equivalence geometry. Specifically, SAGE uses

$$K_{ij}^{\mathrm{soft}} = c + (1 - c) K_{ij}^{\mathrm{bin}},$$
$$K_{ij}^{\mathrm{A}} = \left(K_{ij}^{\mathrm{ling}}\right)^{\alpha} \left(K_{ij}^{\mathrm{soft}}\right)^{\beta}.$$

Here $K_{ij}^{\mathrm{bin}} = 1$ when a_i and a_j are equivalent under the task-specific answer rule, and 0 otherwise. The cross-cluster floor $c \in [0, 1]$ controls how much similarity is retained between non-equivalent answer clusters. Unless otherwise specified, we use $c = 0.8$ in the main experiments and set $\alpha = \beta = 1$. The normalized entropy is used as the group-level uncertainty target:

$$t(G_x) = H_{\mathrm{SAGE}}(G_x).$$

Stage 1. Format supervised fine-tuning. The first training stage teaches the model to follow the output format required for answer and uncertainty extraction. This stage is not intended to calibrate uncertainty. Instead, it standardizes response structure across tasks. For MATH-500, the model learns to output reasoning followed by a boxed answer. For TriviaQA, it learns the explicit `Answer:/Uncertainty:` format. For MMLU-Pro, it learns to end with a final option label. All calibration experiments start from the corresponding format-SFT checkpoint, so that comparisons across targets and training methods are not confounded by formatting failures.

Stage 2. Calibration training. Starting from the format-SFT checkpoint, we train verbal uncertainty using group-level uncertainty targets. For supervised calibration baselines, the target $t(G_x)$ is converted into a target uncertainty expression and optimized with supervised fine-tuning. For preference-based training, we use GUPO, which applies calibration supervision to the verbal uncertainty expression rather than to the full response. All target variants use the same offline response groups, and differ only in how $t(G_x)$ is computed.

For each response $y_i = (z_i, v_i)$, let $u(v_i) \in [0, 1]$ denote the extracted verbal uncertainty. Given $t(G_x)$, we define the calibration score

$$r_i = -\ell\big(u(v_i), t(G_x)\big),$$
$$\ell\big(u(v_i), t(G_x)\big) = \big(u(v_i) - t(G_x)\big)^2 .$$

The scores are normalized within the response group:

$$p_i^{\text{cal}} = \frac{\exp(r_i/T)}{\sum_{j=1}^{K} \exp(r_j/T)}.$$

The resulting preference signal is applied to the uncertainty channel v_i conditioned on the prompt and answer-bearing content z_i. Conceptually, GUPO optimizes

$$\mathcal{L}_{\text{GUPO}} = -\mathbb{E}_{x,G_x}\Bigg[\sum_{i=1}^{K} \text{stopgrad}(p_i^{\text{cal}}) \cdot \log \pi_\theta(v_i \mid x, z_i)\Bigg].$$

In implementation, the calibration loss can be masked to the uncertainty-expression span.

GUPO training hyperparameters

GUPO uses TRL with GUPO-style group preference training and vLLM rollouts. Unless otherwise specified, we use learning rate 1×10^{-6}, cosine scheduling, default linear warmup, and 0.3 epochs. The per-device batch size is 2, with gradient accumulation 8, giving an effective batch size of 16 prompts. Online training uses 8 rollouts per prompt. We use `max_new_tokens` $= 512$, maximum prompt length 384, KL coefficient 0.1, rollout temperature 1.0, and seed 42. All GUPO runs start from the format-SFT checkpoint.

Evaluation protocol. At evaluation time, the trained model generates one response with verbal uncertainty for each validation prompt. We extract the final answer and numerical uncertainty. We report Brier score, Expected Calibration Error, Spearman correlation, and threshold-based accuracy at @80, @60, and @50. Threshold-based accuracy is computed after converting uncertainty into confidence, following the standard high-confidence subset accuracy protocol. Since all methods are evaluated on the same underlying sampled answer pool, raw answer accuracy is not the main comparison. The evaluation instead measures whether verbal uncertainty better ranks reliable examples and better matches the group-level uncertainty target.

C Additional Experiments

C.1 Ablation over Group-Level Targets

We further compare different group-level uncertainty targets under the same GUPO training framework. All variants use the same base model, sampled response groups, training protocol, and evaluation pipeline. The only difference is how the group-level target $t(G_x)$ is computed. This ablation is designed to isolate the effect of target construction from the effect of the optimizer. In other words, all variants receive group rollouts and are trained with the same group-based optimization procedure, but the reward signal is induced by different uncertainty targets.

We compare four targets. **Max Answer Frequency** uses the share of the most frequent extracted answer in the group and converts answer stability into uncertainty. It is directly tied to answer agreement, but it compresses the group into a discrete count and cannot represent graded semantic or reasoning-level differences. **Semantic Entropy** clusters responses into meaning-equivalence classes and computes uncertainty over these clusters. It improves over exact answer counting by merging semantically equivalent generations, but its hard cluster assignments can produce discontinuous and low-resolution targets. **Hard SAGE** uses an answer-aware kernel with $c = 0$, which strictly separates different answer clusters. This maximizes answer separation, but can over-compress cross-answer similarity and collapse the target scale. **SAGE** uses a soft cross-cluster floor with $c = 0.8$, preserving answer-level distinctions while retaining enough cross-cluster similarity to maintain a

Benchmark	Target	Calibration			High-Confidence Accuracy		
		Brier ↓	ECE ↓	Spear. ↑	@80 ↑	@60 ↑	@50 ↑
MMLU-Pro	Maximum Answer Frequency	0.184	0.301	0.210	0.188	0.217	0.240
	Semantic Entropy	0.214	0.316	0.090	0.188	0.183	0.180
	Kernel Language Entropy	0.068	0.205	-0.010	0.163	0.167	0.180
	SAGE	**0.037**	**0.113**	**0.572**	**0.412**	**0.512**	**0.570**
MATH-500	Maximum Answer Frequency	0.155	0.252	**0.515**	0.588	0.683	0.700
	Semantic Entropy	0.190	0.282	0.426	0.575	0.650	0.720
	Kernel Language Entropy	0.0205	0.0348	0.451	0.588	0.700	0.720
	SAGE	**0.0202**	**0.0238**	0.463	**0.613**	**0.717**	**0.740**
TriviaQA	Maximum Answer Frequency	0.158	0.252	0.296	0.502	0.521	0.540
	Semantic Entropy	0.153	0.247	0.404	0.480	0.530	0.540
	Kernel Language Entropy	0.123	0.227	0.268	0.490	0.480	0.520
	SAGE	**0.055**	**0.031**	**0.617**	**0.506**	**0.583**	**0.650**

Table 3: Ablation over group-level uncertainty targets under the same GUPO training framework. SAGE provides the strongest overall calibration signal across benchmarks, especially on calibration error, uncertainty ranking, and low-uncertainty subset accuracy.

c	Target Distribution		Verbal Unc.	Calibration			High-Confidence Accuracy		
	Mean	Std	Mean u_v	Brier ↓	ECE ↓	Spear. ↑	@80 ↑	@60 ↑	@50 ↑
0.0	0.1934	0.2393	0.000	0.0826	0.1779	–	0.562	0.567	0.600
0.1	0.1982	0.2364	0.013	0.0789	0.1712	0.138	0.575	0.583	0.600
0.2	0.2102	0.2295	0.022	0.0775	0.1764	0.333	0.625	0.667	0.660
0.3	0.2275	0.2202	0.065	0.0640	0.1517	0.304	0.625	0.667	0.640
0.4	0.2492	0.2091	0.100	0.0531	0.1363	0.411	0.588	0.683	0.700
0.5	0.2751	0.1970	0.164	0.0371	0.1035	**0.524**	0.600	0.683	0.700
0.6	0.3049	0.1845	0.197	0.0350	0.0969	0.430	0.588	**0.717**	**0.740**
0.7	0.3388	0.1724	0.306	0.0234	0.0326	0.429	0.588	0.700	**0.740**
0.8	0.3770	0.1620	0.347	0.0202	0.0238	0.463	0.613	**0.717**	**0.740**
0.9	0.4201	0.1549	0.427	**0.0188**	**0.0149**	0.466	**0.637**	0.683	0.680
1.0	0.4700	0.1544	0.497	0.0205	0.0348	0.451	0.588	0.700	0.720

Table 4: Sensitivity of SAGE to the soft cross-cluster floor c on MATH-500. Here $c = 0$ corresponds to hard answer separation, while $c = 1$ reduces the target to the pure linguistic kernel. The highlighted row denotes the default setting used in our main experiments.

usable reward scale.

Table 3 reports the results. The comparison reveals several important patterns.

First, all non-SAGE targets show a mismatch between one desirable property and another. Semantic Entropy achieves moderate Spearman correlation and threshold accuracy, but its calibration errors remain high, with Brier score 0.190 and ECE 0.282. This indicates that semantic clustering can provide some ranking signal, but the resulting uncertainty values are poorly calibrated in absolute scale. Max Answer Frequency performs better than Semantic Entropy on Spearman correlation and high-confidence subset accuracy, reaching the best Spearman score among the non-SAGE variants. This is expected because answer frequency is closely tied to final-answer agreement. However, its Brier score and ECE remain substantially worse than SAGE, showing that answer frequency alone is too coarse to provide well-calibrated uncertainty values.

Second, the hard answer-aware variant confirms that answer awareness alone is not sufficient. Hard SAGE with $c = 0$ has much lower Brier and ECE than Semantic Entropy and Max Answer Frequency, but this improvement is misleading because the verbal uncertainty channel nearly collapses, with mean uncertainty only 0.005. The near-zero mean uncertainty also explains why Spearman correlation is weak. A hard block-diagonal answer kernel separates different answer clusters too aggressively, causing the target distribution to lose a

usable scale for training. This supports our claim that the target must be answer-aware, but it must also be scale-preserving.

Third, SAGE with a soft cross-cluster floor provides the best overall trade-off. It achieves the lowest Brier score 0.019 and ECE 0.022, while also obtaining the best high-confidence subset accuracy across @50, @60, and @80. Its Spearman correlation 0.513 is comparable to Max Answer Frequency, but with far better calibration error. This is important because Spearman mainly measures ranking, while Brier and ECE measure whether the numerical uncertainty values are calibrated. SAGE therefore does not merely rank examples better; it also places uncertainty values on a more appropriate scale.

Overall, this ablation supports the central claim of our method. Group rollouts are useful only when the target extracted from the group provides a strong training signal. Max Answer Frequency is answer-faithful but too coarse. Semantic Entropy is more semantic but limited by hard clustering. Hard SAGE is answer-aware but collapses the uncertainty scale. Soft SAGE combines the useful properties of these targets: it preserves answer distinctions, remains smooth through the linguistic kernel, and avoids collapse through the cross-cluster floor. The resulting target gives GUPO a substantially more informative calibration signal.

C.2 Sensitivity to the Cross-Cluster Floor

SAGE uses a soft cross-cluster floor c to control how much similarity is retained between responses with different extracted answers. When $c = 0$, the answer kernel becomes a hard block-diagonal kernel, which strongly separates different answer clusters but may collapse the target scale. When $c = 1$, the answer kernel becomes uninformative and SAGE reduces to the pure linguistic kernel. Intermediate values allow SAGE to preserve answer-level distinctions while retaining enough cross-cluster similarity to provide a usable training signal.

Table 4 reports the full sensitivity study on MATH-500. As c increases, the target mean increases from 0.1934 to 0.4700, while the target standard deviation decreases from 0.2393 to 0.1544. This indicates that the soft floor gradually relaxes hard answer separation and moves the target toward the smoother linguistic-kernel regime. The hard setting $c = 0$ collapses the verbal uncertainty channel, producing $\text{mean } u_v = 0.000$ and no valid Spearman correlation. Soft floors substantially improve Brier score and ECE, with the strongest calibration performance around $c = 0.8$ to $c = 0.9$. These results support the design choice that answer-aware geometry should be introduced softly rather than as hard answer separation.

D Mathematical Analysis of Target Identifiability and Reward Signal

D.1 Why Response-Level Supervision is Insufficient

Let x be a prompt and let the model induce a response distribution

$$p_\theta(y \mid x).$$

Each response y_i contains answer-bearing content z_i, an extracted answer

$$a_i = A(z_i),$$

and a verbal uncertainty value $u_i \in [0, 1]$. The goal of verbal uncertainty alignment is to make u_i reflect the model's uncertainty over possible responses to the same prompt. Thus, the ideal uncertainty target is not a function of a single sampled response, but a functional of the response distribution:

$$t^\star(x) = T(p_\theta(\cdot \mid x)).$$

Equivalently, with a sampled group

$$G_x = \{y_1, \ldots, y_K\}, \qquad y_i \sim p_\theta(\cdot \mid x),$$

we estimate

$$t(G_x) \approx t^\star(x).$$

Under-identification of a single response. A response-level method observes only one sample y for each prompt. We show that this is insufficient because the same observed response can be generated from different underlying response distributions with different uncertainty.

Consider two answer distributions over $\{A, B, C, D\}$:

$$p_1(a \mid x) = (0.90, 0.04, 0.03, 0.03),$$

$$p_2(a \mid x) = (0.30, 0.25, 0.25, 0.20).$$

Suppose the sampled response in both cases gives answer A. From the perspective of a response-level objective, the observed instance is identical:

$$(x, y, A).$$

However, the correct uncertainty target should be very different. For example, under an entropy-based uncertainty target, the distribution p_1 should receive lower uncertainty than p_2, because its probability mass is more concentrated on one answer. Thus the same observed answer A should receive low uncertainty under p_1 but much higher uncertainty under p_2.

This means any pointwise target of the form

$$r(x, y, u)$$

cannot identify the correct distributional uncertainty, because

$$r(x, y, u; p_1) = r(x, y, u; p_2)$$

whenever the method only observes the same (x, y, u). The missing information is the distribution of alternative plausible outputs.

Consequence for SFT and DPO. SFT minimizes a response-level loss:

$$\mathcal{L}_{\mathrm{SFT}} = -\log p_\theta(y, u^\star \mid x).$$

This objective can imitate a provided uncertainty string $u^\star$, but it does not reveal whether y came from a stable distribution or an unstable one.

Similarly, DPO-style methods compare two responses:

$$\mathcal{L}_{\mathrm{DPO}} = -\log \sigma\left(\beta\left[\log \frac{\pi_\theta(y^+ \mid x)}{\pi_{\mathrm{ref}}(y^+ \mid x)} - \log \frac{\pi_\theta(y^- \mid x)}{\pi_{\mathrm{ref}}(y^- \mid x)}\right]\right).$$

The preference pair (y^+, y^-) can indicate which response is preferred, but it does not determine the full uncertainty distribution over possible generations. Many distributions can produce the same pair:

$$(y^+, y^-) \subset \mathrm{supp}(p_\theta(\cdot \mid x)),$$

while having different uncertainty targets

$$T(p_\theta(\cdot \mid x)).$$

Therefore, response-level supervision is under-identified for verbal uncertainty calibration. It can learn local uncertainty patterns, but it cannot determine whether the uncertainty should reflect a concentrated or dispersed response distribution.

D.2 Why Existing Group-Level Targets Give Weak Rewards

Group rollouts address the distributional challenge by exposing multiple samples:

$$G_x = \{y_1, \ldots, y_K\}.$$

However, the target computed from G_x must also be useful for optimization. Suppose the training reward is

$$R(u, G_x) = -\left(u - t(G_x)\right)^2,$$

where $t(G_x)$ is the group-level uncertainty target. If $t(G_x)$ is coarse, discontinuous, or misaligned with answer disagreement, then the reward provides weak or misleading supervision.

Maximum answer frequency is coarse. The maximum answer frequency stability score is

$$s_{\mathrm{MAF}}(G_x) = \max_a \frac{1}{K} \sum_{i=1}^{K} \mathbb{I}[a_i = a].$$

Its uncertainty version is

$$t_{\mathrm{MAF}}(G_x) = 1 - s_{\mathrm{MAF}}(G_x).$$

Let

$$n_a = \sum_{i=1}^{K} \mathbb{I}[a_i = a].$$

Then

$$s_{\mathrm{MAF}}(G_x) = \frac{1}{K} \max_a n_a.$$

Thus MAF depends only on the largest answer count and discards the rest of the group structure.

For example, with $K = 10$,

$$G_1 = (6A, 4B),$$
$$G_2 = (6A, 1B, 1C, 1D, 1E).$$

Both groups receive the same stability score:

$$s_{\mathrm{MAF}}(G_1) = s_{\mathrm{MAF}}(G_2) = 0.6.$$

But G_2 contains more diverse secondary alternatives. Hence the target collapses distinct uncertainty structures into the same reward value when it only uses the largest answer count.

Moreover, MAF can take only K possible stability values:

$$s_{\mathrm{MAF}}(G_x) \in \left\{\frac{1}{K}, \frac{2}{K}, \ldots, 1\right\}.$$

Therefore the reward landscape is discrete. If two groups have the same majority count, then for any uncertainty value u,

$$R(u, G_1) = R(u, G_2),$$

even if their semantic variation or secondary answer disagreement is different. MAF is answer-faithful, but it is too coarse to provide fine-grained calibration supervision.

Semantic Entropy is discontinuous. Semantic Entropy clusters responses into meaning-equivalence classes. Let

$$C(y_i) \in \{1, \dots, M\}$$

be the semantic cluster assignment, and let

$$p_m = \frac{1}{K} \sum_{i=1}^{K} \mathbb{I}[C(y_i) = m].$$

Then

$$H_{\mathrm{SE}}(G_x) = -\sum_{m=1}^{M} p_m \log p_m.$$

The target depends only on hard cluster counts. Therefore, if a response changes slightly but remains in the same cluster,

$$C(y_i) = C(y_i') \quad \Rightarrow \quad H_{\mathrm{SE}}(G_x) = H_{\mathrm{SE}}(G_x').$$

This means the reward is flat within each cluster:

$$\Delta H_{\mathrm{SE}} = 0.$$

However, if a small change moves one response across a cluster boundary, then the empirical cluster distribution changes discontinuously. Suppose one response moves from cluster u to cluster v. Then

$$p_u' = p_u - \frac{1}{K}, \qquad p_v' = p_v + \frac{1}{K}.$$

The entropy change is

$$\begin{aligned}\Delta H_{\mathrm{SE}} = &-\left(p_u - \frac{1}{K}\right) \log \left(p_u - \frac{1}{K}\right) \\ &-\left(p_v + \frac{1}{K}\right) \log \left(p_v + \frac{1}{K}\right) \\ &+ p_u \log p_u + p_v \log p_v.\end{aligned}$$

This jump occurs because the cluster assignment changes, not because the response distribution has changed smoothly. Therefore SE produces a piecewise-constant reward:

$$\text{within cluster: } \Delta H_{\mathrm{SE}} = 0,$$

$$\text{across boundary: } \Delta H_{\mathrm{SE}} \neq 0.$$

As a result, SE improves over exact answer counting for paraphrases, but its hard clustering gives weak credit assignment for fine-grained uncertainty calibration.

KLE is smooth but not answer-faithful. KLE replaces hard clusters with a continuous kernel. Let

$$K_{ij} = k(e(y_i), e(y_j)), \qquad K_{ij} \in [0, 1],$$

and normalize

$$P = \frac{K}{\mathrm{tr}(K)}.$$

The normalized von Neumann entropy is

$$H_{\mathrm{KLE}}(G_x) = -\frac{1}{\log K} \mathrm{tr}(P \log P).$$

This target is smooth because small changes in embeddings produce small changes in pairwise similarities. However, smoothness alone is insufficient. KLE assumes that embedding similarity reflects answer compatibility:

$$K_{ij} \text{ high} \quad \Rightarrow \quad a_i \equiv a_j.$$

This assumption fails in structured answer spaces. For example, multiple-choice labels A, B, C, and D may be close in embedding space even though they are mutually exclusive answers:

$$A \not\equiv B \not\equiv C \not\equiv D.$$

We can see the failure mathematically. Suppose a response group contains K mutually incompatible answers, but generic embeddings assign high similarity to all pairs:

$$K_{ii} = 1, \qquad K_{ij} = r \quad (i \neq j), \qquad r \approx 1.$$

Then

$$K = (1 - r)I + r\mathbf{1}\mathbf{1}^\top.$$

The eigenvalues of K are

$$\lambda_1 = 1 + (K - 1)r, \qquad \lambda_2 = \cdots = \lambda_K = 1 - r.$$

Since $\mathrm{tr}(K) = K$, the eigenvalues of P are

$$\mu_1 = \frac{1 + (K-1)r}{K},$$
$$\mu_2 = \cdots = \mu_K = \frac{1-r}{K}.$$

As $r \to 1$,

$$\mu_1 \to 1, \qquad \mu_2, \ldots, \mu_K \to 0.$$

Therefore,

$$H_{\mathrm{KLE}}(G_x) = -\frac{1}{\log K} \sum_{i=1}^{K} \mu_i \log \mu_i \to 0.$$

So KLE predicts low uncertainty even when all final answers are incompatible. This shows that KLE is smooth, but its generic geometry can underestimate task-critical answer disagreement.

D.3 Summary

The above analysis shows that a useful group-level uncertainty target must satisfy three requirements:

1. **Answer-faithfulness**: the target should reflect task-critical answer agreement and disagreement.

2. **Reward smoothness**: the target should change continuously with meaningful variations in the response group.

3. **Scale preservation**: the target should preserve a usable uncertainty range rather than collapsing to near-zero or near-one values.

MAF is answer-faithful but coarse. SE handles paraphrases but produces flat and discontinuous rewards because of hard clustering. KLE is smooth but can collapse incompatible answers when generic embeddings place them close together. Therefore, repeated sampling is not sufficient by itself: the target must transform sampled responses into an answer-aware and optimization-friendly reward signal.

E Qualitative Case Study

Case Study: When Semantic Similarity Hides Answer Disagreement

Prompt. A multiple-choice question asks the model to select one option from $\{A, B, C, D\}$.

Repeated samples from the model.

$$G_x = \{(A, 0.18),\ (A, 0.21),\ (B, 0.24),\ (C, 0.26),\ (A, 0.19),\ (B, 0.23)\}.$$

Observed answer distribution.

$$p(A) = \frac{3}{6}, \qquad p(B) = \frac{2}{6},$$
$$p(C) = \frac{1}{6}, \qquad p(D) = 0.$$

Interpretation. The model frequently expresses low uncertainty, but its sampled answers are not stable. The group contains mutually exclusive choices, so the appropriate uncertainty target should be higher than the expressed values suggest.

Target	Behavior on this group	Limitation / Effect
MAF	$s_{\mathrm{MAF}} = 3/6 = 0.50$	answer-faithful but coarse
KLE	high similarity among option labels	may underestimate disagreement
SAGE	separates A, B, C as incompatible	answer-aware target

This example illustrates why generic semantic similarity can be misleading for uncertainty calibration. In a multiple-choice task, option labels such as A, B, and C are linguistically similar because they are short symbolic labels with nearly identical surface form. A generic embedding-based target may therefore assign high pairwise similarity to responses ending in different option labels. This can make the response group appear more coherent than it actually is, producing an overly low uncertainty target.

However, from the task perspective, A, B, and C are not nearby semantic variants. They are mutually exclusive decisions. A model that alternates among these choices should express higher uncertainty, even if the generated explanations have similar wording. SAGE corrects this mismatch by evaluating response groups under answer-aware equivalence. Responses with the same extracted option remain close, while responses with different option

labels are separated in the answer-conditioned geometry. As a result, the group receives a higher and more appropriate uncertainty target, which gives GUPO a clearer supervision signal for reducing overconfident verbal expressions.

F Use of AI Assistants

The authors used AI assistants only for language polishing and minor editing of the manuscript text. AI assistants were not used to generate research ideas, design the method, conduct experiments, produce results, or write technical claims. All content, including the proposed method, experimental design, analysis, and final manuscript, was reviewed, verified, and approved by the authors.