TACO: IMPROVING UNCERTAINTY DECOMPOSITION IN BLACK-BOX LLMS VIA TARGET-AWARE CLARIFICATION OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Uncertainty decomposition is essential for trustworthy reasoning with large language models (LLMs). Recent approaches for black-box LLMs uncertainty decomposition leverage input clarifications to construct ensembles over different interpretations of the same input. However, the effectiveness of such methods critically depends on the quality of generated clarification sets. Existing clarification generation approaches typically treat clarification generation as a general text generation task and overlook its target-dependent nature: the same clarification may be interpreted differently by different target LLMs, introducing additional uncertainty and degrading uncertainty decomposition. To address this limitation, we propose **TACO**, a **T**arget-**A**ware **C**larification **O**ptimization framework for black-box LLM uncertainty decomposition. TACO formulates clarification generation as a reinforcement learning problem and directly optimizes the clarification generator using feedback from the target LLM through two complementary reward mechanisms: an uncertainty decomposition reward that improves the separation between ambiguous and clear inputs, and an entropy alignment reward that encourages ambiguity-aware diversity while preserving consistency for clear inputs. Extensive experiments demonstrate that TACO consistently improves uncertainty decomposition performance across diverse target LLMs and tasks.

1 INTRODUCTION

Uncertainty quantification in deep learning is the process of measuring and representing the confidence, reliability, and potential error range of a model's predictions, which often involves separating total predictive uncertainty into aleatoric uncertainty—caused by intrinsic ambiguity in the data—and epistemic uncertainty—resulting from gaps in the model's knowledge or capacity. This separation is referred to as uncertainty decomposition (UD). With the widespread application of large language models (LLMs), UD for LLMs has gained increasing attention He et al. (2025); Walha et al. (2025); Jayasekera et al. (2025).

Advanced LLMs—such as GPT-4o, Claude 3.5 Sonnet, and DeepSeek-V4—are either black-box models with inaccessible parameters or open-weight models whose sheer scale renders local deployment impractical. Consequently, traditional uncertainty decomposition methods that require access to model parameters and training procedures, such as deep ensembles Lakshminarayanan et al. (2017), are difficult to apply. To address this limitation, Hou et al. (2024) recently proposed the Input Clarification Ensemble (ICE), which shifts the ensemble operation from the model side to the input side. Specifically, ICE generates multiple clarifications of an original input. Under the assumption that the clarifications remove all aleatoric ambiguity, any remaining uncertainty in the target LLM's predictions can be attributed to epistemic uncertainty. Semantic entropy over the LLM's predictions on each clarified input then serves as an estimate of epistemic uncertainty, thereby enabling uncertainty decomposition. Thus, the effectiveness of ICE hinges on whether the clarification set can faithfully capture the underlying ambiguities in the original input.

To construct a clarification set, existing methods primarily rely on prompt engineering Ekin (2026) or supervised fine-tuning Pareja et al. (2025). The former directly prompts an LLM to generate clarifications for a given input, whereas the latter uses human-annotated clarification data to train

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

a model that maps the original input to its clarified form. Both approaches, however, overlook the characteristics of the target LLM when building the clarification generator. As a result, a clarification set that performs well for one target LLM may underperform for another. The reason is that when the original input is concatenated with a clarification and fed into the target LLM, the observed semantic entropy conflates the target LLM's epistemic uncertainty about the clarification itself with its epistemic uncertainty about the original input. Since each LLM interprets clarifications differently owing to its distinct knowledge, capturing epistemic uncertainty about the original input requires minimizing the epistemic uncertainty of the clarifications for the target LLM.

Based on these observations, we propose the **Target-Aware Clarification Optimization (TACO)** framework, which trains a clarification generator tailored to the target LLM. As illustrated in Figure 1, we formulate clarification generation as a reinforcement learning problem, where the generator acts as the policy and is updated directly through reward signals from the black-box target LLM.

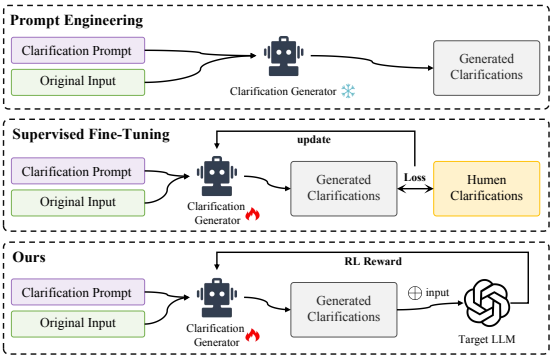

Figure 1: **Comparison of clarification generation.** Unlike prompt engineering (top: frozen generator, no optimization) and supervised fine-tuning (middle: trained on human-annotated clarifications), our TACO framework (bottom) optimizes the clarification generator via reinforcement learning using reward signals from the black-box target LLM.

Specifically, we propose two complementary rewards that jointly optimize the clarification generator from distinct perspectives. The first is an **Uncertainty Decomposition Reward**, which directly optimizes the clarification generator such that the resulting clarification set maximizes the target LLM's AUROC over mixed batches of ambiguous and clear samples. The rationale is that the target LLM's epistemic uncertainty about the clarifications manifests as noise that obscures the separation between ambiguous and unambiguous samples; maximizing AUROC therefore naturally drives this uncertainty toward zero. The second is an **Entropy Alignment Reward**, which leverages semantic entropy of the generated clarifications as feedback to ensure that the clarification generator captures the underlying ambiguities in ambiguous inputs without degrading clear inputs. Specifically, this reward encourages the generator to produce clarifications that exhibit high entropy for ambiguous inputs while maintaining consistent semantics for clear ones. Through this target-aware optimization mechanism, TACO generates clarification sets that not only effectively expose the intrinsic ambiguity of the input, but also align closely with the target LLM's capabilities.

Extensive experiments show that TACO consistently outperforms recent baselines on black-box LLM uncertainty decomposition. Further analysis reveals that TACO produces more effective clarification sets, improves the discrimination between aleatoric and epistemic uncertainty, and enhances the robustness of uncertainty decomposition across diverse target LLMs.

2 Related Work

Uncertainty quantification aims to assess the reliability of model predictions. Early methods rely on token-level probabilities Malinin & Gales (2018); Maddox et al. (2019); Jiang et al. (2021); Shen et al. (2023), while recent studies estimate semantic uncertainty through sampling Manakul et al. (2023); Chen & Mueller (2024); Huang et al. (2025) or input perturbations Gao et al. (2024); Marjanovic et al. (2024); Kuhn et al. (2023); Farquhar et al. (2024); Nikitin et al. (2024). However, these methods mainly capture total uncertainty and cannot distinguish ambiguity-induced uncertainty from model limitations, motivating uncertainty decomposition.

Classical uncertainty decomposition separates uncertainty into aleatoric and epistemic components under Bayesian models Neal (2012); Kendall & Gal (2017), but is difficult to apply to black-box LLMs. ICE Hou et al. (2024) addresses this by generating multiple clarifications and attributing prediction variation across these clarifications to aleatoric uncertainty, while treating residual un-

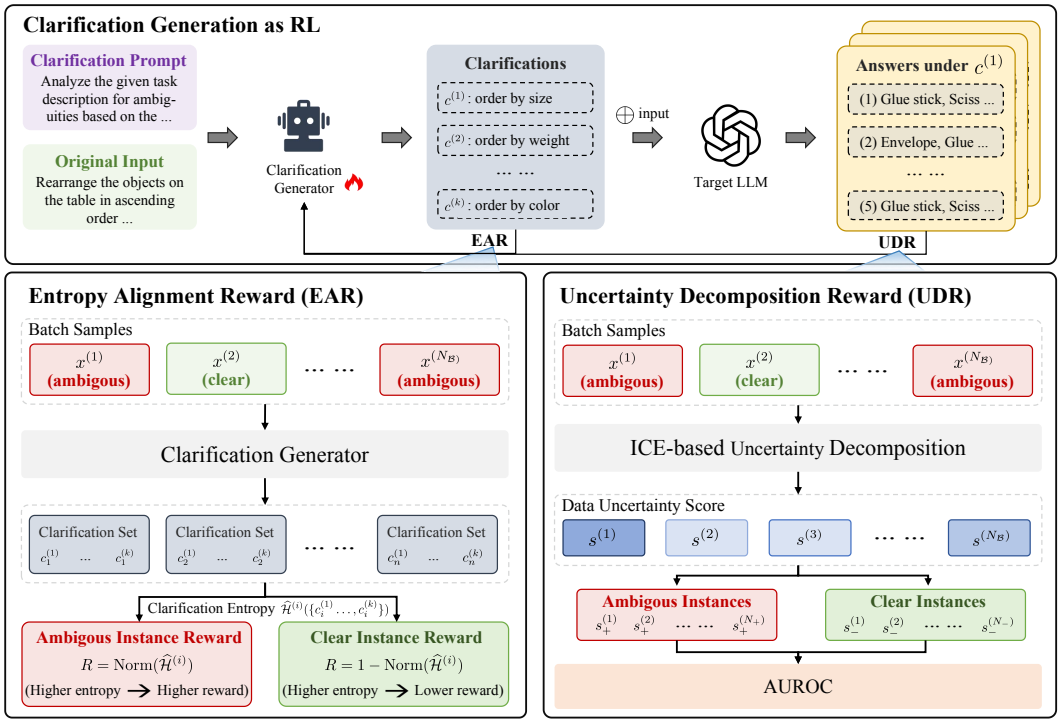

Figure 2: Overview of TACO, our proposed framework for target-aware clarification optimization.

certainty as epistemic uncertainty. Subsequent works improve aggregation Walha et al. (2025) and extend ICE to ICL Jayasekera et al. (2025), but their effectiveness still relies heavily on the quality of clarification sets, while the problem of optimizing clarification generation itself remains relatively underexplored.

Existing clarification set generators—whether based on prompting or supervised fine-tuning Hou et al. (2024); Walha et al. (2025); Jayasekera et al. (2025)—do not explicitly adapt to target LLMs, and may introduce additional epistemic uncertainty for the target LLM on the generated clarifications. In contrast, we formulate clarification generation as a reinforcement learning problem and optimize the generator using feedback from the target LLM, thereby aligning clarification sets with the target LLM's capabilities. **More related work is provided in Appendix A.**

3 Methodology

3.1 Background: ICE-based Uncertainty Decomposition

ICE-based UD estimates uncertainty caused by input ambiguity without modifying the target model. The key idea is to generate multiple plausible clarifications of the same input and analyze the variation of the target model's predictions under these different interpretations.

Given an input X, a clarification generator M_C samples K texts, $c^{(k)}$, from $q_\phi(C \mid X)$:

$$c^{(k)} \sim q_\phi(C \mid X), \tag{1}$$

where $c^{(k)}$ represents a reasonable set of clarifications for X.

Then, ICE-based methods aggregate the predictive distributions of the target model under different clarifications:

$$q(Y \mid X) = \mathbb{E}_{C \sim q_\phi(C|X)} \left[q(Y \mid X \oplus C) \right]. \tag{2}$$

Based on the conditional mutual information identity, the total uncertainty of the ensemble predictive distribution can be decomposed as:

$$\mathcal{H}(q(Y \mid X)) = \underbrace{\mathcal{I}(Y; C \mid X)}_{\text{aleatoric}} + \underbrace{\mathbb{E}_{q(C|X)}\left[\mathcal{H}(q(Y \mid X \oplus C))\right]}_{\text{epistemic}}, \tag{3}$$

where mutual information term quantifies prediction variation across clarifications. Large variation indicates that different interpretations of the input lead to different predictions, reflecting aleatoric uncertainty induced by input ambiguity. The second term measures the uncertainty that remains after a specific interpretation has been provided, and therefore reflects uncertainty associated with the target model's limited knowledge or capabilities.

However, this decomposition assumes that clarifications only resolve input ambiguity without introducing extra uncertainty. In practice, different LLMs may interpret the same clarification differently, causing ICE-estimated epistemic uncertainty to include clarification-induced uncertainty. Thus, clarification generators should be optimized for the target LLM to minimize this effect.

3.2 Clarification Generation as Reinforcement Learning

To generate clarification sets tailored to a specific target LLM, we formulate clarification generation as a reinforcement learning problem that seeks an optimal set of discrete clarifications C^* for a given input X, clarification model M_C, target model M_T, and dataset D. Each clarification is a token sequence defined over the vocabulary V of the clarification model, with a maximum sequence length L.

The clarification model M_C is an LLM that generates a set of K candidate clarifications $C = c^{(1)}, c^{(2)}, \ldots, c^{(K)}$, conditioned on the input X and a clarification-generation prompt. We treat this combined input as the state s. The complete clarification prompt is provided in Figure x. For each candidate k, the clarification model generates a token sequence autoregressively according to $M_C(c_l^{(k)} \mid s, c_{<l}^{(k)})$ where $c_l^{(k)}$ denotes the l-th token of the k-th clarification. Once the candidate set C has been generated, it is evaluated using the reward function R. The resulting training objective is:

$$\max_{M_C} R(\{M_T(X, c^{(k)}\}_{k=1}^K), y),\ c^{(k)} \sim \prod_{l=1}^{L} M_C(c_l^{(k)} | s, c_{<l}^{(k)}) \tag{4}$$

During training, we update the parameters of the clarification model M_C, aiming to enable it to learn a policy for generating comprehensive and effective clarifications for the given input X and target model M_T.

3.3 Reward Function Design

Unlike supervised fine-tuning, which relies on predefined clarification references, TACO evaluates clarifications according to their influence on the target LLM. Specifically, we design two complementary rewards from both global and local perspectives. The first reward evaluates whether the generated clarification set enables effective uncertainty decomposition across a batch. The second reward aligns the entropy induced by clarifications with the intrinsic ambiguity of individual inputs.

Uncertainty Decomposition Reward We use the batch-level AUROC of data uncertainty scores as the uncertainty decomposition reward. This reward encourages the clarification generator to produce clarification sets that better distinguish ambiguous from clear inputs. Since clarification-induced epistemic uncertainty introduces noise into uncertainty decomposition, maximizing AUROC guides the generator to reduce such uncertainty while preserving ambiguity-related uncertainty.

For each input $x^{(i)}$ in the batch, we compute its data-uncertainty score as:

$$s^{(i)} = [\widehat{\mathcal{H}}(\{M_T^{(g)}(x^{(i)} \oplus c_k^{(i)})\}_{k=1,g=1}^{K,G}) - \frac{1}{K}\sum_{k=1}^{K} \widehat{\mathcal{H}}(\{M_T^{(g)}(x^{(i)} \oplus c_k^{(i)})\}_{g=1}^{G})]_+, \tag{5}$$

where $c_k^{(i)}$ is the k-th clarification generated by M_C, K is the number of clarifications, and $M_T^{(g)}$ denotes the g-th stochastic response sampled from M_T, with G amples drawn for each clarification. The first term estimates the total uncertainty over responses generated under all clarifications,

whereas the second term estimates the average uncertainty remaining under each individual clarification. Here, $\widehat{\mathcal{H}}$ denotes empirical entropy, and $[z]_+ = \max(0, z)$ ensures a non-negative score.

Following prior work Hanley & McNeil (1982), we compute the batch-level AUROC using the data-uncertainty scores of all instances in the batch:

$$\text{AUC}_{\mathcal{B}} = \frac{1}{N_+ N_-} \sum_{i=1}^{N_+} \sum_{j=1}^{N_-} [\mathbb{I}(s_+^{(i)} > s_-^{(j)}) + \frac{1}{2}\mathbb{I}(s_+^{(i)} = s_-^{(j)})], \tag{6}$$

where $+$ and $-$ denote ambiguous and clear instances in the batch, respectively. N is the number of samples. This metric measures how well data uncertainty separates ambiguous inputs from clear ones.

Because an AUROC of 0.5 corresponds to random discrimination, we subtract this baseline and normalize the resulting value to obtain the batch-level reward:

$$R_{\text{UD}}^{(i)} = \text{Norm}(\text{AUC}_{\mathcal{B}} - 0.5) \tag{7}$$

where $R_{\text{UD}}^{(i)} \in [0, 1]$ denotes the AUROC-based uncertainty decomposition reward assigned to each instance in the batch, and $\text{Norm}(\cdot)$ denotes a normalization function. Because this reward is computed at the batch level, all instances within the same batch receive the same AUROC-based reward.

Entropy Alignment Reward Although the uncertainty decomposition reward optimizes clarification generation from a batch perspective, it does not explicitly enforce whether each individual clarification set captures the ambiguity structure of its corresponding input. Therefore, we introduce an entropy alignment reward to provide instance-level supervision.

For each input instance $x^{(i)}$, let $C_i = \{c_i^{(k)}\}_{k=1}^{K}$ denote the clarification set generated by the policy model M_C. We first compute the empirical entropy [] of the clarification set C_i:

$$\widehat{\mathcal{H}}^{(i)} = \widehat{\mathcal{H}}(C_i), \tag{8}$$

The instance-level reward is then defined as

$$R_{\text{EA}}^{(i)} = \begin{cases} \text{Norm}(\widehat{\mathcal{H}}^{(i)}), & x^{(i)} \in \mathcal{X}_+, \\ 1 - \text{Norm}(\widehat{\mathcal{H}}^{(i)}), & x^{(i)} \in \mathcal{X}_-, \end{cases} \tag{9}$$

where $\mathcal{X}_+$ and $\mathcal{X}_-$ denote the sets of ambiguous and clear instances, respectively. $R_{\text{EA}}^{(i)} \in [0, 1]$ evaluates the quality of the clarification set generated for input $x^{(i)}$.

The intuition is that clarifications should align the target model's uncertainty with the intrinsic ambiguity of the input. For ambiguous inputs, effective clarifications should preserve multiple interpretations and induce higher entropy, while for clear inputs, they should maintain semantic consistency and avoid artificial uncertainty. Thus, the entropy alignment reward encourages the generator to capture genuine ambiguity while reducing clarification-induced noise.

3.4 Policy Optimization

Given the two rewards above, we optimize the clarification generator with proximal policy optimization (PPO) Schulman et al. (2017). Let π_θ denote the trainable policy induced by M_C, and let $\pi_{\theta_{\text{old}}}$ denote the policy used to sample the current batch of clarification sets. For each training instance $x^{(i)}$, we combine the batch-level and instance-level feedback into a single scalar reward:

$$R^{(i)} = R_{\text{UD}}^{(i)} + R_{\text{EA}}^{(i)}, \tag{10}$$

Since each clarification is a discrete token sequence, we apply PPO at the token level. For the l-th token of the k-th clarification generated for $x^{(i)}$, the probability ratio between the updated and old policies is

$$\rho_{i,k,l}(\theta) = \frac{\pi_\theta(c_{i,l}^{(k)} \mid s_i, c_{i,<l}^{(k)})}{\pi_{\theta_{\text{old}}}(c_{i,l}^{(k)} \mid s_i, c_{i,<l}^{(k)})}. \tag{11}$$

Table 1: Performance comparison of UD methods on AmbigInst, AmbigQA, and GSM8K. AmbigInst and AmbigQA evaluate ambiguity detection using aleatoric uncertainty, while GSM8K evaluates correctness prediction using total uncertainty. GLM-4-9B is used as backbone for clarification generator, and LLaMA-3-8B and Qwen-2.5-7B are used as target LLMs. The best results are highlighted in bold.

LLaMA-3-8B						
Methods	**AmbigInst**		**AmbigQA**		**GSM8K**	
	AUROC	Best-F1	AUROC	Best-F1	AUROC	Best-F1
Ensembles	68.86 ± 3.86	76.32 ± 3.73	55.11 ± 4.16	57.40 ± 5.22	84.46 ± 2.81	82.85 ± 2.97
Semantic	65.73 ± 1.36	62.45 ± 1.56	55.86 ± 1.38	51.48 ± 1.11	80.71 ± 1.51	81.34 ± 1.08
Ask4Conf	56.99 ± 1.34	72.58 ± 1.06	52.49 ± 1.39	59.89 ± 1.15	64.00 ± 1.77	73.71 ± 1.31
ICE	79.54 ± 2.18	76.78 ± 2.31	67.10 ± 1.76	71.73 ± 1.77	85.25 ± 2.83	86.52 ± 2.82
Spectral Uncertainty	80.33 ± 2.29	79.35 ± 2.15	69.08 ± 1.39	69.23 ± 1.07	87.86 ± 2.15	89.67 ± 2.07
TACO (Ours)	**90.49** ± 2.44	**89.33** ± 2.83	**75.01** ± 3.35	**79.23** ± 3.77	**91.12** ± 2.03	**92.85** ± 2.12
Qwen-2.5-7B						
Methods	**AmbigInst**		**AmbigQA**		**GSM8K**	
	AUROC	Best-F1	AUROC	Best-F1	AUROC	Best-F1
Ensembles	69.53 ± 3.55	75.48 ± 3.38	63.99 ± 3.12	59.34 ± 4.01	87.62 ± 2.29	87.07 ± 2.23
Semantic	67.78 ± 1.32	73.61 ± 1.86	59.90 ± 1.23	57.85 ± 1.38	84.88 ± 1.10	87.59 ± 1.66
Ask4Conf	55.78 ± 1.56	64.75 ± 1.59	56.98 ± 1.54	59.33 ± 1.24	59.61 ± 1.26	77.11 ± 1.34
ICE	83.07 ± 2.49	80.09 ± 2.67	68.69 ± 1.48	75.30 ± 1.43	89.03 ± 2.54	91.13 ± 2.45
Spectral Uncertainty	83.80 ± 2.14	82.77 ± 2.27	70.50 ± 1.48	75.68 ± 1.59	89.04 ± 2.41	91.69 ± 2.63
TACO (Ours)	**87.75** ± 2.84	**87.69** ± 2.79	**73.96** ± 3.66	**80.00** ± 3.95	**94.13** ± 2.18	**94.11** ± 1.16

We estimate the advantage as

$$A^{(i)} = R^{(i)} - V_\psi(s_i), \tag{12}$$

where V_ψ is a value function used as the baseline. The same sequence-level advantage is assigned to all tokens in the generated clarification set for $x^{(i)}$, because the reward is computed after the target LLM evaluates the complete set of clarifications. The clipped PPO objective is then defined as

$$\mathcal{L}_{\text{PPO}}(\theta) = \mathbb{E}_{i,k,l}\Big[\min(\rho_{i,k,l}(\theta)A^{(i)}, \text{clip}(\rho_{i,k,l}(\theta), 1-\epsilon, 1+\epsilon)A^{(i)})\Big], \tag{13}$$

where ϵ is the clipping threshold. This objective constrains each update to remain close to the sampling policy, which stabilizes learning under sparse and delayed feedback from the target LLM.

4 Experiments

4.1 Experimental Setup

Detailed descriptions of comparison baselines, backbone models (including clarification and target LLMs), prompt design, and implementation details are provided in Appendix B.

Dataset Following prior work Hou et al. (2024); Walha et al. (2025), we evaluate TACO on two complementary tasks. **Ambiguity detection** tests whether estimated data uncertainty is higher for ambiguous inputs than for clear ones. We use **AmbigQA** Min et al. (2020), which contains naturally ambiguous open-domain questions with multiple valid interpretations, and **AmbigInst** Hou et al. (2024), which pairs ambiguous instructions with clarifications and associated input-output examples. **Correctness prediction** tests whether estimated total uncertainty is higher for incorrect LLM predictions than for correct ones. We use **GSM8K** Cobbe et al. (2021), a benchmark of grade-school math word problems, to assess this relationship in mathematical reasoning.

Evaluation Protocol To ensure a fair comparison with prior work on uncertainty decomposition Hou et al. (2024), we follow the same evaluation protocol and measure uncertainty estimation performance using AUROC and Best-F1. AUROC measures the discriminative ability of uncertainty scores

Table 2: Ablated results on AmbigInst and AmbigQA. LLaMA-3-8B is used as the target LLM, while GLM-4-9B or Gemma-1.1-7B-IT is used as the backbone for clarification generator. The best results are highlighted in bold.

GLM-4-9B				
Methods	**AmbigInst**		**AmbigQA**	
	Auroc	Best-F1	Auroc	Best-F1
Zero-Shot	79.54	76.78	52.38	54.87
ICL	83.09	79.91	54.60	58.29
SFT	85.07	84.77	67.10	71.73
TACO (Ours)	**90.49**	**89.33**	**75.01**	**79.23**
TACO w/o EAR	88.42	86.21	72.84	76.90
TACO w/o UDR	85.83	88.89	70.56	72.22

Gemma-1.1-7B-IT				
Methods	**AmbigInst**		**AmbigQA**	
	Auroc	Best-F1	Auroc	Best-F1
Zero-Shot	54.85	36.18	53.91	57.15
ICL	59.12	43.93	58.69	65.93
SFT	89.96	88.00	66.90	69.33
TACO (Ours)	**92.39**	**90.79**	**71.87**	**72.80**
TACO w/o EAR	91.63	90.20	69.25	71.67
TACO w/o UDR	89.05	87.57	65.32	68.54

between ambiguous and clear samples for ambiguity detection, or between correct and incorrect predictions for correctness prediction, with higher values indicating better uncertainty discrimination. Best-F1 measures the maximum classification performance achieved by selecting the optimal threshold over uncertainty scores, reflecting the effectiveness of uncertainty-based decisions.

4.2 Main Results

The results are presented in Table 1, where GLM-4-9B serves as the backbone for the clarification generator, while LLaMA-3-8B and Qwen-2.5-7B are used as target LLMs to assess whether TACO generalizes effectively across different target models.

Performance on Ambiguity Detection We evaluate ambiguity detection on the AmbigInst and AmbigQA datasets. Across both datasets and target models, TACO consistently achieves the best performance among all methods. Specifically, on AmbigInst with LLaMA-3-8B, TACO attains an AUROC of 90.49 and a Best-F1 of 89.33, outperforming the strongest baseline by 10.16 and 9.98 percentage points, respectively. With Qwen-2.5-7B, TACO reaches 87.75 AUROC and 87.69 Best-F1, yielding improvements of 3.95 and 4.92 points. Similar gains are observed on AmbigQA.

Performance on Correctness Prediction We evaluate correctness prediction on GSM8K, which tests whether the estimated total uncertainty score can effectively distinguish correct from incorrect model predictions. TACO again achieves the best performance with both target LLMs. Specifically, with LLaMA-3-8B, TACO attains an AUROC of 91.12 and a Best-F1 of 92.85, surpassing the strongest baseline by 3.26 and 3.18 percentage points, respectively. With Qwen-2.5-7B, TACO reaches 94.13 AUROC and 94.11 Best-F1, corresponding to improvements of 5.09 and 2.42 points.

Overall, the results across different datasets and backbones demonstrate the generalization of TACO, highlighting the importance of optimizing the clarification generator with target-LLM feedback for ICE-based uncertainty decomposition in LLMs.

4.3 Ablation Study

We conduct an ablation study to evaluate the contributions of the two reward components, UDR and EAR, in our method. To this end, we construct two variants: ***w/o EAR***, which removes EAR from the full model, and ***w/o UDR***, which removes UDR from the full model. We also include several clarification generation methods as baselines, including Zero-Shot, In-Context Learning, and Supervised Fine-Tuning (SFT). The results are presented in Table 2.

First, fine-tuned methods generally outperform Zero-Shot and ICL, while *w/o UDR* performs similarly to SFT across datasets and backbones. EAR regulates the clarification generator's prediction entropy to encourage diversity for ambiguous inputs and avoid unnecessary ambiguity for clear ones. Like SFT, it relies on ambiguity labels without target-LLM feedback, which may explain their similar performance.

Second, *w/o EAR* substantially outperforms SFT, suggesting that UDR drives the main performance gains. UDR incorporates target-LLM feedback through uncertainty-based ranking of ambiguous

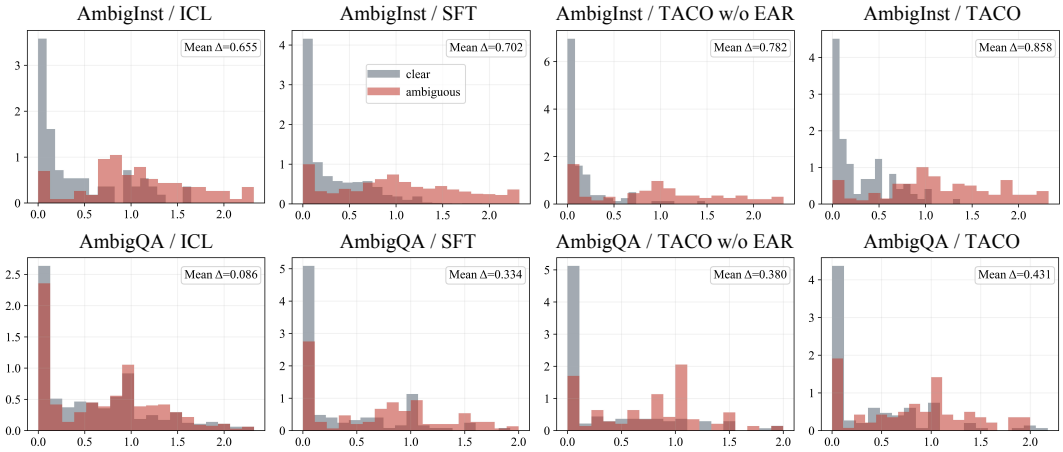

Figure 3: Data uncertainty distributions for clear and ambiguous samples on AmbigInst and AmbigQA, using GLM-4-9B as the clarification generator and LLaMA-3-8B as the target LLM. Mean Δ denotes the average uncertainty gap (ambiguous minus clear). Larger values indicate greater separation between the two groups.

Table 3: AUROC, data and model uncertainty on AmbigInst and AmbigQA. GLM-4-9B and Gemma-1.1-7B-IT serve as clarification generators, with LLaMA-3-8B as the target LLM.

GLM-4-9B					Gemma-1.1-7B-IT				
Dataset	**Method**	**AUROC**	**Data UQ**	**Model UQ**	**Dataset**	**Method**	**AUROC**	**Data UQ**	**Model UQ**
AmbigInst	ICL	83.09	0.724	0.656	AmbigInst	ICL	59.12	0.217	0.926
	SFT	85.07	0.707	0.493		SFT	89.96	0.663	0.467
	TACO	**90.49**	0.779	**0.478**		TACO	**92.39**	0.864	**0.369**
AmbigQA	ICL	54.60	0.645	0.700	AmbigQA	ICL	58.69	1.780	0.742
	SFT	67.10	0.500	0.518		SFT	66.90	0.943	0.738
	TACO	**75.01**	0.567	**0.440**		TACO	**71.87**	1.089	**0.724**

and clear inputs, aligning clarification generation with the target model. Full TACO consistently outperforms both ablations across datasets and backbones, indicating that EAR and UDR provide complementary benefits for uncertainty decomposition.

4.4 Further Analysis

Necessity of Target-Aware Optimization We evaluate uncertainty decomposition across three target LLMs using the same human-annotated clarification set. Figure 4 shows substantial performance differences on AmbigQA: AUROC scores are 90.56, 85.35, and 75.43 for LLaMA-3-8B, Qwen-2.5-7B, and DeepSeek-R1-8B, respectively, with a cross-model variance of 39.39. Best-F1 shows a similar trend, and performance gaps persist on AmbigInst despite being smaller. These results show that identical clarifications yield different decomposition performance across target LLMs, motivating target-aware optimization of the clarification generator.

Analysis of Clarification-Induced Model Uncertainty To assess whether clarifications introduce spurious uncertainty into ICE, we compare ICL, SFT, and TACO under matched datasets, generator backbones, target LLM, and inference settings (Table 3). Model uncertainty, measured by the target LLM's mean semantic entropy under clarifications, should reflect model limitations rather than clarification-induced noise. With GLM-4-9B, TACO reduces model uncertainty relative to SFT from 0.493 to 0.478 on AmbigInst and from 0.518 to 0.440 on AmbigQA, while increasing data uncertainty from 0.707 to 0.779 and from 0.500 to 0.567, respectively. It also yields lower model uncertainty than ICL (0.656 and 0.700) and achieves AUROCs of 90.49 and 75.01. The same pattern with Gemma-1.1-7B-IT suggests that TACO preserves input ambiguity while reducing clarification-related noise across generator backbones.

Table 4: Clarification quality on AmbigInst and AmbigQA. IC is the mean pairwise BERTScore among clarifications (lower indicates greater diversity); CI is the BERTScore between clarifications and the input (higher indicates greater relevance); NR is the proportion of "unknown" target-LLM responses. GLM-4-9B is the clarification generator and LLaMA-3-8B is the target LLM.

AmbigInst					
Methods	**Clear**		**Ambiguous**		**NR**
	IC	**CI**	**IC**	**CI**	
ICL	0.825	0.774	0.838	0.783	0%
SFT	0.818	0.781	0.828	0.784	0%
TACO w/o EAR	0.780	0.764	0.792	0.765	0%
TACO (Ours)	0.787	0.792	0.782	0.763	0%

AmbigQA					
Methods	**Clear**		**Ambiguous**		**NR**
	IC	**CI**	**IC**	**CI**	
ICL	0.083	0.026	0.038	0.013	7.6%
SFT	0.656	0.293	0.736	0.364	5.4%
TACO w/o EAR	0.615	0.300	0.625	0.371	3.8%
TACO (Ours)	0.630	0.310	0.614	0.366	4.3%

Analysis of Data Uncertainty Separation Figure 3 reveals that TACO's gains stem from structurally reshaping the data uncertainty distribution. Both TACO and its variants consistently outperform ICL and SFT, showing that UDR alone, through target model feedback, effectively improves the separation between ambiguous and clear samples. With the additional instance-level constraints from EAR, the full TACO further increases Mean Δ from 0.782/0.380 to 0.858/0.431, indicating that EAR enhances the distributional margin beyond global ranking optimization. This calibration enables uncertainty scores to better reflect intrinsic input ambiguity rather than clarification-induced noise, leading to consistent improvements in AUROC and Best-F1.

Analysis of Clarification Quality Table 4 evaluates clarification diversity (IC), input relevance (CI), and unknown-response rate (NR). On AmbigQA, ICL has low IC (0.083/0.038 for clear/ambiguous inputs) but very low CI (0.026/0.013) and high NR (7.6%), suggesting that its diversity often comes from irrelevant or hard-to-interpret clarifications. SFT improves relevance and NR but has higher IC for ambiguous than clear inputs (0.736 vs. 0.656); TACO w/o EAR retains this mismatch (0.625 vs. 0.615) despite reducing NR to 3.8%. Full TACO reverses the IC pattern on both AmbigQA (0.614 vs. 0.630) and AmbigInst (0.782 vs. 0.787), while maintaining relevance and a 4.3% NR. This pattern is consistent with EAR adapting clarification diversity to input ambiguity. Beyond the quantitative evaluation, we conduct a case study to qualitatively analyze the clarification behaviors of different methods by comparing their generated clarifications. **Additional analysis is provided in Appendix C**.

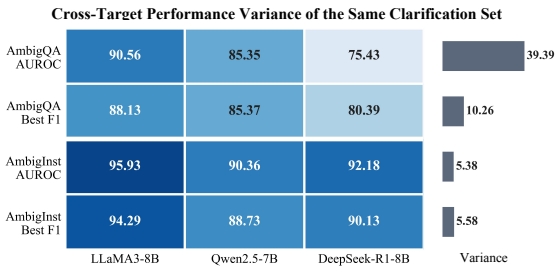

Figure 4: Performance of the human-annotated clarification set on AmbigQA and AmbigInst with three target LLMs: LLaMA-3-8B, Qwen-2.5-7B, and DeepSeek-R1-8B.

5 Conclusion

In this work, we identify a key limitation of existing clarification generation approaches, which serve as a central component of uncertainty decomposition for black-box LLMs: they treat clarification generation as a general text generation task while overlooking its target-dependent nature. To address this limitation, we propose TACO, a target-aware clarification optimization framework that formulates clarification generation as a reinforcement learning problem and directly optimizes the generator using feedback from the target LLM. By incorporating uncertainty decomposition and entropy alignment rewards, TACO learns clarification sets that are better aligned with the target LLM, enabling more reliable separation of ambiguity-induced uncertainty (data uncertainty) from model uncertainty while mitigating uncertainty introduced by clarifications. Extensive experiments across diverse target LLMs and tasks demonstrate that TACO consistently improves uncertainty decomposition performance, validating the effectiveness of target-aware clarification optimization for black-box LLMs.

6 Ethics Statement

This work adheres to the ICLR Code of Ethics. In this study, no human subjects or animal experimentation was involved. All datasets used, including AmbigInst, AmbigQA and GSM8K, were sourced in compliance with relevant usage guidelines, ensuring no violation of privacy. We have taken care to avoid any biases or discriminatory outcomes in our research process. No personally identifiable information was used, and no experiments were conducted that could raise privacy or security concerns. We are committed to maintaining transparency and integrity throughout the research process.

7 Reproducibility statement

To support reproducibility, we provide detailed descriptions of our method and experimental setup, including dataset, training procedures, model configurations, and hardware specifications. Our code will be made publicly available after the paper is published to facilitate replication and verification of the results.

8 LLM Usage

We used large language models (LLMs) solely for language editing, including sentence rephrasing, grammar checking, and improvements to clarity and readability. All research ideas, methods, experimental designs, and analyses were developed or conducted by the authors without LLM assistance. The authors take full responsibility for the manuscript, including LLM-assisted text, and have ensured that its content complies with ethical guidelines and does not involve plagiarism or scientific misconduct.

References

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural network. In *International conference on machine learning*, pp. 1613–1622. PMLR, 2015.

Jiuhai Chen and Jonas Mueller. Quantifying uncertainty in answers from any language model and enhancing their trustworthiness. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5186–5200, 2024.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Sabit Ekin. Prompt engineering for chatgpt: a quick guide to techniques, tips, and best practices. *Journal of Engineering Technology*, 43(1), 2026.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630, 2024.

Stanislav Fort, Huiyi Hu, and Balaji Lakshminarayanan. Deep ensembles: A loss landscape perspective. *arXiv preprint arXiv:1912.02757*, 2019.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *International Conference on Machine Learning*, pp. 1050–1059, 2016.

Xiang Gao, Jiaxin Zhang, Lalla Mouatadid, and Kamalika Das. Spuq: Perturbation-based uncertainty quantification for large language models. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2336–2346, 2024.

Team Glm, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Dan Zhang, Diego Rojas, Guanyu Feng, Hanlin Zhao, et al. Chatglm: A family of large language models from glm-130b to glm-4 all tools. *arXiv preprint arXiv:2406.12793*, 2024.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

James A Hanley and Barbara J McNeil. The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, 143(1):29–36, 1982.

Wenchong He, Zhe Jiang, Tingsong Xiao, Zelin Xu, and Yukun Li. A survey on uncertainty quantification methods for deep learning. *ACM Computing Surveys*, 2025.

Bairu Hou, Yujian Liu, Kaizhi Qian, Jacob Andreas, Shiyu Chang, and Yang Zhang. Decomposing uncertainty for large language models through input clarification ensembling. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 19023–19042, 2024.

Yuheng Huang, Jiayang Song, Zhijie Wang, Shengming Zhao, Huaming Chen, Felix Juefei-Xu, and Lei Ma. Look before you leap: An exploratory study of uncertainty analysis for large language models. *IEEE Transactions on Software Engineering*, 51(2):413–429, 2025.

Binyuan Hui, Jian Yang, Zeyu Cui, Jiaxi Yang, Dayiheng Liu, Lei Zhang, Tianyu Liu, Jiajun Zhang, Bowen Yu, Keming Lu, et al. Qwen2. 5-coder technical report. *arXiv preprint arXiv:2409.12186*, 2024.

I. Shavindra Jayasekera, Jacob Si, Filippo Valdettaro, Wenlong Chen, Aldo A. Faisal, and Yingzhen Li. Variational uncertainty decomposition for in-context learning. In *Advances in Neural Information Processing Systems*, 2025.

Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. How can we know when language models know? on the calibration of language models for question answering. *Transactions of the Association for Computational Linguistics*, 9:962–977, 2021.

Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? *Advances in neural information processing systems*, 30, 2017.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=VD-AYtP0dve.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Wesley J Maddox, Pavel Izmailov, Timur Garipov, Dmitry P Vetrov, and Andrew Gordon Wilson. A simple baseline for bayesian uncertainty in deep learning. *Advances in neural information processing systems*, 32, 2019.

Andrey Malinin and Mark Gales. Predictive uncertainty estimation via prior networks. *Advances in neural information processing systems*, 31, 2018.

Potsawee Manakul, Adian Liusie, and Mark Gales. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pp. 9004–9017, 2023.

Sara Marjanovic, Isabelle Augenstein, and Christina Lioma. Investigating the impact of model instability on explanations and uncertainty. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 11854–11879, 2024.

Sewon Min, Julian Michael, Hannaneh Hajishirzi, and Luke Zettlemoyer. Ambigqa: Answering ambiguous open-domain questions. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pp. 5783–5797, 2020.

Radford M Neal. *Bayesian learning for neural networks*, volume 118. Springer Science & Business Media, 2012.

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

Aldo Pareja, Nikhil Shivakumar Nayak, Hao Wang, Krishnateja Killamsetty, Shivchander Sudalairaj, Wenlong Zhao, Seungwook Han, Abhishek Bhandwaldar, Guangxuan Xu, Kai Xu, et al. Unveiling the secret recipe: A guide for supervised fine-tuning small llms. In *International Conference on Learning Representations*, volume 2025, pp. 73565–73597, 2025.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Maohao Shen, Yuheng Bu, Prasanna Sattigeri, Soumya Ghosh, Subhro Das, and Gregory Wornell. Post-hoc uncertainty learning using a dirichlet meta-model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 9772–9781, 2023.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D Manning. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 5433–5442, 2023.

Nassim Walha, Sebastian G. Gruber, Thomas Decker, Yinchong Yang, A. Javanmardi, Eyke Hüllermeier, and Florian Buettner. Fine-grained uncertainty decomposition in large language models: A spectral approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025.

A Related Work

A.1 Uncertainty Quantification

Uncertainty quantification aims to assess the reliability of model predictions. Early studies Malinin & Gales (2018); Maddox et al. (2019); Jiang et al. (2021); Shen et al. (2023) primarily focused on classification models, where uncertainty could be estimated from token-level entropy. However, LLMs generate open-ended text, making token-level uncertainty measures insufficient for capturing semantic correctness: responses with different lexical realizations may convey identical meanings, whereas lexically similar responses may differ substantially in validity. Consequently, recent studies Kuhn et al. (2023); Farquhar et al. (2024); Nikitin et al. (2024) have shifted the focus from token-level uncertainty to semantic uncertainty. Existing approaches can be broadly categorized into two categories: sampling-based methods Manakul et al. (2023); Chen & Mueller (2024); Huang et al. (2025), which measure uncertainty by comparing multiple responses generated for the same input, and perturbation-based methods Gao et al. (2024); Marjanovic et al. (2024), which evaluate prediction stability under different input variations. While these methods provide effective estimates of total predictive uncertainty, they remain unable to determine whether the uncertainty originates from inherent ambiguity in the input or from limitations in the model's own knowledge and reasoning capabilities.

A.2 Uncertainty Decomposition for LLMs

Classical uncertainty decomposition Kendall & Gal (2017) is primarily developed under the bayesian neural network (BNN) Neal (2012), where model parameters are treated as random variables induced by the training data. Under this formulation, prediction variability across different parameter samples represents epistemic uncertainty, whereas the residual uncertainty in the predictive distribution given a fixed parameter sample corresponds to aleatoric uncertainty. However, the

true posterior distribution over model parameters is intractable, practical approaches typically rely on approximations, such as variational inference Blundell et al. (2015), MC dropout Gal & Ghahramani (2016), or deep ensembles Lakshminarayanan et al. (2017).

With the emergence of LLMs, these approaches become increasingly difficult to apply, as model parameters and posterior distributions are typically inaccessible in black-box settings. To address this limitation, Hou et al. (2024) proposed Input Clarification Ensembling (ICE), which shifts the ensemble operation from the model side to the input side by generating multiple plausible clarifications of an input and measuring the variation in the target LLM's predictions across these interpretations. The prediction variability induced by different clarifications is attributed to aleatoric uncertainty, while the remaining uncertainty after clarification is regarded as an approximation of epistemic uncertainty. Building upon ICE, Walha et al. (2025) introduced a finer-grained semantic aggregation strategy to improve the disentanglement of the two uncertainty components. More recently, Jayasekera et al. (2025) extended this paradigm to in-context learning (ICL) by constructing auxiliary queries to approximate Bayesian uncertainty decomposition. Despite these advances, the effectiveness of ICE-based methods critically depends on the quality of the generated clarification sets, an aspect that remains largely underexplored.

A.3 Clarification Generation for Uncertainty Decomposition in LLMs

Existing approaches Hou et al. (2024); Walha et al. (2025); Jayasekera et al. (2025) for constructing clarification sets can be broadly categorized into two paradigms: prompt engineering and supervised fine-tuning (SFT). Prompt-based methods rely on carefully designed instructions to guide an LLM to generate multiple plausible interpretations of an input. Although flexible and requiring no additional training, these methods generate clarifications according to the intrinsic capability and behavior of the clarification model itself, without considering how a specific target LLM interprets the generated clarifications. Consequently, a clarification set that performs well for one target LLM may introduce substantial uncertainty when applied to another target LLM.

SFT-based approaches learn clarification generation from human-annotated clarification datasets, enabling more consistent and controllable outputs. However, these methods primarily optimize the correspondence between generated clarifications and human-written references, rather than the uncertainty decomposition performance of the downstream target LLM. A clarification that is semantically reasonable from a human perspective may still be suboptimal for a particular target LLM, as it can introduce additional epistemic uncertainty about the clarification itself and obscure the uncertainty caused by the original input ambiguity.

Therefore, existing methods prioritize the linguistic quality or human preference of clarifications, while overlooking the interaction between the clarification generator and the target LLM. In contrast, we formulate clarification generation as a reinforcement learning problem, where the generator is updated via reward signals from the target LLM. Through this target-aware mechanism, our clarification sets are generated to align with the target LLM's capabilities, thereby minimizing its epistemic uncertainty about the clarifications and improving uncertainty decomposition performance.

B Additional Experimental Setup

B.1 Backbone LLMs

Throughout our experiments, we use GLM-4-9B Glm et al. (2024) and Gemma-1.1-7B-IT Team et al. (2024) as clarification generators. These models are responsible for generating clarification sets and are used to evaluate the robustness of TACO across different generator backbones. We employ LLaMA-3-8B Grattafiori et al. (2024), Qwen-2.5-7B Hui et al. (2024), and DeepSeek-R1-8B Guo et al. (2025) as target LLMs to evaluate the performance of uncertainty decomposition across different target models. During both training and evaluation, the target LLMs remain frozen and are used solely to generate responses for uncertainty estimation, reward computation, and cross-model comparisons of clarification effectiveness.

B.2 Implementation Details

B.2.1 Experimental Configurations

In the main experiments, we evaluate the overall performance of our uncertainty decomposition framework. To control for the effect of the clarification generator, we fix GLM-4-9B as the clarification generator and use LLaMA-3-8B and Qwen-2.5-7B as the target LLMs, respectively. TACO is trained separately for each target LLM, ensuring that the clarification generator receives feedback from the same target model used during evaluation, which allows us to assess whether TACO generalizes consistently across different target models. For baseline methods that do not require a clarification generator, uncertainty decomposition is performed directly using responses sampled from the corresponding target LLM, ensuring a fair comparison.

In the ablation study, we evaluate different clarification optimization strategies within the ICE-based uncertainty decomposition framework. To isolate the effect of the clarification generator, we fix LLaMA-3-8B as the target LLM and use GLM-4-9B and Gemma-1.1-7B-IT as the clarification generators. By varying the generator backbone while keeping the target model unchanged, we examine the robustness and generalizability of TACO across different clarification generators. In all experiments, the target LLM remains frozen and is used only to generate responses for uncertainty estimation and reward computation.

B.2.2 Dataset Construction

We conduct experiments on AmbigInst, AmbigQA, and GSM8K. For AmbigInst, we randomly sample 100 examples from the full dataset, including 50 clear and 50 ambiguous instances, to construct the training set; all remaining examples are used for testing. For AmbigQA, we randomly select 100 examples—50 clear and 50 ambiguous—from the original training split for training. Following the evaluation protocol of ICE, we use 200 examples from the validation split as the test set. For GSM8K, we similarly sample 100 examples from the original training split for training and evaluate on 200 examples from the validation split, consistent with the ICE evaluation protocol.

B.2.3 Training Details

We optimize TACO using proximal policy optimization (PPO) and update only the LoRA parameters of the clarification generator. The LoRA rank is set to 16, with $\alpha = 32$ and a dropout rate of 0.05. The target LLM remains frozen throughout training and is used only to produce responses for uncertainty estimation and reward computation. For the ambiguity detection tasks on AmbigInst and AmbigQA, we train for 100 epochs and construct each batch with a balanced number of clear and ambiguous examples. The batch size is set to 4 for both datasets, and the learning rate is 2×10^{-6}. For GSM8K, the batch size is set to 2 and the learning rate is 1×10^{-5}.

B.2.4 Inference Details

During clarification generation, the policy samples two candidate outputs for each input, from which we extract and retain up to 10 valid clarifications. Each clarification is then combined with the original input and passed to the target LLM multiple times to estimate the corresponding predictive distribution. For each clarified input, we sample five target-model responses. Target-model responses are generated with a temperature of 0.5 and top-$p = 0.9$.

B.3 Prompt Design

To ensure reproducibility and fair comparison, we adopt the same clarification and reasoning prompts as ICE Hou et al. (2024), as shown in Figure 7, 8 and 9.

B.4 Baselines

We compare TACO with five representative baselines, covering ensemble-based uncertainty estimation, semantic uncertainty, confidence elicitation, and clarification-based uncertainty decomposition.

- **Ensembles*** Hou et al. (2024); Fort et al. (2019) approximates model ensembling in the black-box setting by aggregating the predictive distributions obtained using multiple sets of in-context demonstrations. Following prior work, the superscript $*$ indicates that this method does not train independent model instances and therefore differs from standard Deep Ensembles.
- **Semantic Entropy** Kuhn et al. (2023) samples multiple responses from the target LLM, groups semantically equivalent responses, and computes the entropy over the resulting semantic clusters as an estimate of total predictive uncertainty.
- **Ask4Conf** Tian et al. (2023) directly elicits uncertainty from the target LLM by first obtaining its prediction and then asking the model to report its confidence in the correctness of that prediction. For ambiguity detection, we adapt the confidence prompt to ask whether the input is ambiguous.
- **Input Clarification Ensembling (ICE)** Hou et al. (2024) generates multiple plausible clarifications of an input and evaluates the target LLM under the corresponding interpretations. Prediction disagreement across clarifications is used to estimate aleatoric uncertainty, while the average uncertainty remaining under each clarification is treated as epistemic uncertainty. The original clarification generator is constructed using prompting or supervised fine-tuning, without explicitly optimizing it for the target LLM.
- **Spectral Uncertainty** Walha et al. (2025) is a recent extension of the ICE framework that employs a finer-grained semantic aggregation strategy to improve the separation of aleatoric and epistemic uncertainty.

All methods are evaluated using the same datasets, target LLMs, and evaluation metrics. For clarification-based methods, we use the same clarification-model backbone whenever applicable. All reported experimental results are averaged over five random seeds. The experiments are conducted on a Linux-based system equipped with seven NVIDIA A800 80G GPUs.

C Further Analysis

C.1 Case Study

We present one clear case and one ambiguous case from AmbigQA to qualitatively examine how different methods regulate clarification diversity. These examples illustrate that desirable clarification behavior is inherently input-dependent. For a clear input, the generated clarifications should remain semantically consistent with the original question. For an ambiguous input, they should cover distinct yet relevant interpretations. An effective clarification generator should therefore neither maximize nor minimize diversity indiscriminately, but instead adapt clarification diversity to the intrinsic ambiguity of each input.

C.1.1 Clear Case Analysis

As shown in Figure 5, because the original question has a single intended interpretation, its data uncertainty should be close to zero. However, the clarifications generated by ICL exhibit substantial semantic drift, introducing unrelated bands and songs such as “System of a Down” and “She’s Come Undone”. This drift inflates the data-uncertainty score to 1.922. The resulting 13 unique answers therefore reflect spurious ambiguity introduced by the clarifications rather than the intrinsic ambiguity of the input.

SFT substantially improves clarification relevance, achieving a “CI” BERTScore of 0.717. Its “IC” BERTScore of 0.874 also indicates stronger convergence for this clear input. Nevertheless, although SFT preserves the original band and song, it introduces an unnecessary distinction between the lead and backup vocalists. This role inconsistency produces multiple answer variants and thus still overestimates the data uncertainty.

TACO w/o EAR further reduces data uncertainty to 0.013 by using target-model feedback, demonstrating the effectiveness of UDR in suppressing unnecessary response variation. However, some clarifications refer to another Nickelback song, “Someday”. This observation shows that low re-

sponse variation alone does not guarantee semantic faithfulness, since the target model may produce the same answer even when the clarification has drifted from the original input.

TACO incorporates EAR and achieves the highest "IC" BERTScore of 0.925, while consistently preserving both the original song and the requested vocalist role. Consequently, all target-model responses converge to "Chad Kroeger", yielding a single unique answer and zero data uncertainty. This case demonstrates that UDR reduces unnecessary variation in target-model responses, whereas EAR prevents semantic drift and redundant interpretive diversity. Together, they enable TACO to avoid overestimating uncertainty for clear inputs.

C.1.2 Ambiguous Case Analysis

As shown in Figure 6, the queried president changes over time. The question should therefore exhibit high data uncertainty when its missing temporal condition is properly clarified. ICL obtains a data-uncertainty score of 1.355, but one of its clarifications still refers to an unspecified year. Because the temporal condition remains unresolved, part of the uncertainty associated with the input ambiguity may instead be attributed to model uncertainty, leading to an underestimated data-uncertainty score.

SFT generates two nearly identical clarifications, indicating insufficient exploration of the plausible temporal interpretations. This limited ambiguity coverage results in an excessively low data-uncertainty score of 0.437 and substantially underestimates the intrinsic ambiguity of the input.

In TACO w/o EAR, UDR uses target-model feedback to identify time as the relevant ambiguity dimension, increasing data uncertainty to 1.500. However, two clarifications both refer to the current president, introducing redundancy and limiting the coverage of distinct temporal interpretations.

TACO generates complementary clarifications for 2015, 2021, and the current time. This relevant and non-redundant coverage yields the lowest "IC" BERTScore and the highest data-uncertainty score of 2.000. This case demonstrates that UDR helps identify ambiguity dimensions that affect the target model, whereas EAR regulates instance-level clarification diversity and reduces redundant interpretations. Their complementary effects allow TACO to cover plausible interpretations more comprehensively and avoid underestimating the intrinsic ambiguity of ambiguous inputs.

#1 Question: Who sings with nickelback in she keeps me up ? **Label: Clear**

Clarifications Generated by Different Methods

ICL
1. Asking for the lead singer of System of a Down. (Unrelated)
2. Asking for the co-lead singer of System of a Down. (Unrelated)
3. Asking who sang the vocals for "She's Come Undone" released in 1969. (Unrelated)
...

SFT
1. Asking for the lead singer for Nickelback on "She Keeps Me Up". (Right)
2. Asking for the backup vocalist with Nickelback on "She Keeps Me Up". (Unrelated)
3. Asking for the lead vocalist of "She Keeps Me Up" by Nickelback. (Right) ...

TACO w/o EAR
1. Asking for the lead singer of Nickelback's song "Someday". (Right)
2. Asking for the guest vocalist featured in Nickelback's song "Someday". (Unrelated)
3. Asking for the lead singer for Nickelback in "She Keeps Me Up". (Right) ...

TACO
1. Asking for the lead vocalist of Nickelback's "She Keeps Me Up". (Right)
2. Asking for the main vocalist performing "She Keeps Me Up" by Nickelback. (Right)

Response Generated by Target LLM

ICL
1. Serj Tankian
2. Serj Tankian
3. Dolores O'Riordan; Karen Carpenter; Unknown; Aimee Mann. ...

SFT
1. Chad Kroeger
2. Avril Lavigne
3. Chad Kroeger
...

TACO w/o EAR
1. Chad Kroeger
2. Chad Kroeger; Sara Bareilles
3. Chad Kroeger
...

TACO
1. Chad Kroeger
2. Chad Kroeger

Limitations

A lot of spurious ambiguity leads to overestimated data uncertai-nty.

A few spurious ambiguity remains.

Enables relatively consistent responses from the target model but lacks semantic constraints on the clarification text.

UDR and EAR work together to prevent TACO from overestimating uncertainty for clear inputs.

Related Metrics

Methods	Data Ucertainty	IC BERTScore	CI BERTScore	Unique Answers
ICL	1.922	0.546	0.486	13
SFT	0.208	0.874	0.717	3
TACO w/o EAR	0.013	0.739	0.625	2
TACO	0.000	0.925	0.688	1

Figure 5: A clear case from AmbigQA. "IC BERTScore" measures the semantic similarity among clarifications, where a higher score indicates greater convergence. "CI BERTScore" measures the semantic relevance of the clarifications to the original input, where a higher score indicates stronger input consistency. "Unique Answers" denotes the number of distinct responses produced by the target LLM.

#2 Question: Who is the president of miami dade college? **Label: Ambiguous**

Clarifications Generated by Different Methods

ICL

1. Asking for the current president of Miami Dade College. (Right)
2. Asking for the person who has served as president of Miami Dade College since an unspecified year. (Invalid)
3. Asking for the president of Miami Dade College in 2020. (Right) ...

SFT

1. Asking for the current president of Miami Dade College. (Right)
2. Asking for the president of Miami Dade College as of 2021. (Repetion)
3. Asking for the 2021 president of Miami Dade College. (Repetition)
...

TACO w/o EAR

1. Asking for the person currently holding the office of president at Miami Dade College. (Repetion)
2. Asking for the president of Miami Dade College at the start of the 2022 academic year. (Right)
3. Asking for the president of Miami Dade College as of the current date. (Repetion)
...

TACO

1. Asking for the president of Miami Dade College as of 2021. (Right)
2. Asking for the current president of Miami Dade College. (Right)
3. Asking for the president of Miami Dade College as of 2015. (Right)
...

Response Generated by Target LLM

ICL

1. Jeffrey S. Hoch; Dr. Juan F. Carlos Rodriguez; Eduardo J. Padrón
2. J. Mario Martínez; J. Donald Cohen; Jeffrey S. Hoch; J. David Armstrong Jr.
3. Eduardo J. Padrón; Eugenia B. Thomas ...

SFT

1. Jeffrey S. Hoch; Dr. Juan F. Cartagena Jr.
2. Dr. Juan F. Carta; Jeffrey S. Hoch; Jeffrey S. Linkon; Dr. Madeline Pumariega
3. Jeffrey S. Hoch; Dr. Mark B. Rosenberg
...

TACO w/o EAR

1. Dr. Juan F. Carlos Toirac; Jeffrey S. Hoch
2. Dr. Juan F. Carlos Alfonso/Rhodes; Dr. Juan F. Carta; Dr. Juan F. Carlos Rodriguez; Eliana Fernández
3. Dr. Juan F. Carlos Torres; Eliana Reyes; Eliana Fernández; Dr. Juan M. Echavarría
...

TACO

1. Dr. Juan F. Carlos Rodríguez; Dr. Juan F. Carlos Alfonso
2. Dr. Juan F. Mejia; Jeffrey S. Hoch; Dr. Juan F. Alfonso; Jeffrey S. Linkon
3. Eliana Fernández; Elián Cepero; Ehren Rosenberg; Eugenia B. Tomaszewski
...

Limitations

Invalid clarifications remain.

The ambiguity space is underexplored.

Ambiguity remains underexplored, but response diversity exceeds SFT, due to weak clarification constraints.

UDR and EAR help TACO cover plausible interpretations and avoid underestimating ambiguity.

Related Metrics

Methods	Data Ucertainty	IC BERTScore	CI BERTScore	Unique Answers
ICL	1.355	0.836	0.882	21
SFT	0.437	0.884	0.880	6
TACO w/o EAR	1.500	0.838	0.805	12
TACO	2.000	0.769	0.871	14

Figure 6: A ambiguous case from AmbigQA. “IC BERTScore” measures the semantic similarity among clarifications, where a higher score indicates greater convergence. “CI BERTScore” measures the semantic relevance of the clarifications to the original input, where a higher score indicates stronger input consistency. “Unique Answers” denotes the number of distinct responses produced by the target LLM.

AmbigInst

Clarification Prompt

Objective
Analyze the given task description for ambiguities based on the description itself and the provided input question. If the task description is ambiguous, your task is to clarify it by interpreting the ambiguous concepts, specifying necessary conditions, or using other methods. Provide all possible disambiguations.

Important Rules
1. Perform detailed analyses before concluding whether the task description is clear or ambiguous.
2. Output disambiguations in the specified format.
3. Some seemingly unambiguous task descriptions are actually ambiguous given that particular input. So, do not forget to leverage the input to analyze whether the task description is underspecified.

Output Format
Your output should follow this format:
Analyses:
[Think step-by-step to reason on the clarity of the task description. After that, output your judgement on whether the task description is ambiguous or not]

Disambiguations:
1. [One disambiguated task description]
2. [Another disambiguated task description]
3. [Yet another disambiguated task description]
...

If the task description is clear and unambiguous, simply output:
Disambiguations:
1. No clarification needed.

Inference Prompt

[Sample Instruction] [Sample Input]

Figure 7: Overview of the clarification and inference prompts used for AmbigInst. The clarification prompts in the upper panel are provided to the clarification model to identify and resolve ambiguities, whereas the inference prompts in the lower panel are used by the target model to answer the clarified questions.

AmbigQA

Clarification Prompt

In what follows, you will be given some questions that might be ambiguous. These ambiguities can arise from various factors, including but not limited to:

1. Ambiguous references to entities in the question.
2. Multiple properties of objects/entities in the question leading to different interpretations.
3. Ambiguities due to unclear timestamps.
4. Ambiguities stemming from unclear locations.
5. Multiple valid answer types based on the question.

For each question, you are to provide at least two distinct rephrasings that resolve these ambiguities. By "rephrasing," we mean you should reformulate the question to be clear and direct, eliminating any possible ambiguity without altering the original intent of the question. You should not seek further information or produce a binary (yes-no) question as a result of the clarification. Instead, you must create a direct question (wh-question) that aims to obtain a specific answer.

Please format your responses as follows (with at least two rephrasings per question):
Clarifications:
1. [First rephrased question]
2. [Second rephrased question]
3. [Third rephrased question]
...

If the original question is already clear and unambiguous, you should indicate this by stating, "No clarification needed."

Now follow the given examples and clarify the question.

Inference Prompt

[Demonstrations]
[Sample Instruction] [Sample Input]

Figure 8: Overview of the clarification and inference prompts used for AmbigQA.

GSM8K

Clearification Prompt

In this task, you will receive a single question, and your goal is to generate multiple versions of it that convey the same meaning as the original. Please format your responses as follows:
Rephrase 1: [Your rephrased question]
Rephrase 2: [Another rephrased question]
Rephrase 3: [Yet another rephrased question]
....

Here are two examples:
Question: John has 3 boxes. Each box is 5 inches by 6 inches by 4 inches. The walls are 1 inch thick. What is the total inner volume of all 3 boxes?

Rephrase 1: What is the combined internal volume of the three boxes that John has, considering each box measures 5 inches by 6 inches by 4 inches with 1-inch thick walls?
Rephrase 2: If John owns three boxes, and each box has dimensions of 5 inches by 6 inches by 4 inches with 1-inch thick walls, what is the total inner volume of these boxes?
Rephrase 3: How much space is there inside all three of John's boxes when you account for their dimensions of 5 inches by 6 inches by 4 inches with walls that are 1 inch thick?

Inference Prompt

[Demonstrations]
[Qusetion]

Figure 9: Overview of the clarification and inference prompts used for GSM8K.