Label-Confidence-Aware Uncertainty Quantification in Natural Language Generation

Qinhong Lin[a], Yinglun Feng[a], Yuhao Zhang[a], Zhongliang Yang[b,a,*], Linna Zhou[b,a,*]
[a]School of Cyberspace Security, Beijing University of Posts and Telecommunications, Beijing, China
[b]QuanCheng Laboratory, Jinan, China
{greenred99, yangzl, zhoulinna}@bupt.edu.cn

***Abstract*—Large Language Models (LLMs) demonstrate remarkable capabilities in generative tasks but pose potential risks due to their tendency to generate hallucinatory responses. Therefore, Uncertainty Quantification (UQ), which aims to distinguish the validity of answers, is crucial for ensuring the safety and robustness of AI systems. However, existing methods primarily rely on measuring the entropy of multiple stochastic samples to represent uncertainty, often overlooking the specific uncertainty information associated with the candidate answer under evaluation. This oversight can lead to biased classification outcomes. In this paper, we investigate the discrepancy between global entropy from multiple samples and local confidence of candidate answer, and propose a Label-Confidence-Aware Uncertainty Quantification (LCA-UQ) method based on Pointwise Kullback-Leibler (PKL) divergence. Our method effectively bridges the gap between the consistency of sampled outputs and the calibration of the candidate answer, thereby enhancing the reliability and stability of uncertainty assessments. Empirical evaluations across a range of popular LLMs and NLP datasets reveal that label sources significantly impact classification. Furthermore, our approach effectively captures the nuances between sampling results and label sources, demonstrating superior performance in uncertainty estimation.**

***Index Terms*—uncertainty, KL-Divergence, LLMs, self-consistency, confidence**

I. Introduction

Large language models (LLMs) have demonstrated formidable capabilities in natural language processing tasks such as machine translation [1], abstract text summarization [2], and question-answering [3]. Techniques such as In-context Learning (ICL) [4] and Chain-of-Thought (CoT) [5] have further enhanced model performance on complex reasoning tasks and scenarios involving unseen data, consistently setting new benchmarks. However, despite their proficiency under scaling laws [6], these models underperform on more challenging tasks like mathematical problems [7]. A significant concern is that, rather than refusing to answer, models are more likely to generate answers that include illusory reasoning processes and hallucinations. Uncertainty estimation and measurement have become essential tools aiding in determining the extent to which humans can trust AI-generated content. Previous works in this field have involved prompting LLMs to self-assess the confidence of their own answers or employing confidence or entropy assessments based on model outputs. *Semantic Entropy* (SE) [8] has introduced semantic-based entropy prediction schemes in that account for the synonym phenomena inherent in language models, performing answer aggregation in semantic space. *SAR* [9] and *MARS* [10] focus on more precisely measuring about the information content based on semantic importance weighting to offer viable approaches to align the sampling entropy more closely with the actual value. LUQ [11] and CotUQ [12] focus on long generation scenarios, further expanding the application scope of UQ. In existing works, single-sample-based approaches fail to capture the weight of samples within the overall sampling distribution, resulting in its high dependence on calibration. Although approaches that use consistency across multiple samples for uncertainty quantification (UQ) have shown strong results, a common issue in current research is the gap between consistent information in the sampling space and individual calibration. As shown in Fig. 1, when given a question, in the beam search multi-sampling strategy, three out of the five answers generated by the LLM are correct, but due to the high overall entropy value, the LLM may be marked as unable to answer this question. Such an error arises because the entropy threshold used in the evaluation considers only the absolute value, such as the common $-\log(0.5)$, while disregarding the model's own distribution for the question. Specifically, the greedy decoding probability is lower than the probability corresponding to the sample entropy value, which is 0.1661, as shown below. We further quantitatively analyze this issue as well as the fragility of existing UQ methods in the Preliminary section.

To mitigate this issue, in this work, we propose a label-confidence-aware uncertainty quantification (LCA-UQ) method, as shown in Fig. 1, based on Pointwise KL-Divergence (PKL) to bridge the gap between self-consistency information and calibration information. Specifically, we first sample answers from the model's distribution space, as well as the output probabilities to calculate the self-consistency and entropy of the sample set. We then convert the entropy into the Gibbs probability, representing the overall probability of the model's sampling space. Next, we select a candidate answer, such as the greedy decoding answer, to represent the model's response to the question and obtain its probability. Finally, we use the Pointwise Kullback-Leibler divergence (PKL) between the two information as a measure of uncertainty, to classify whether the model can answer the question and whether the answer can be trusted. The LCA-UQ method can be integrated with Monte Carlo and semantic clustering on any single-sample confidence evaluation scheme, establishing its flexibility. We conducted experiments using multiple different entropy measurement

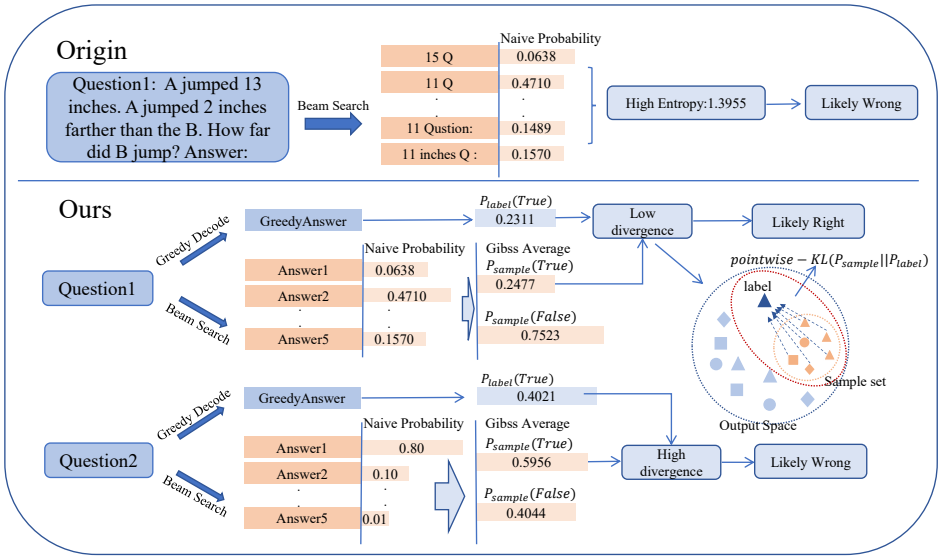


Fig. 1: Ignoring the probability information of the label answer in free-form may lead to incorrect uncertain classification. We term it as label confidence unawareness, and integrate the omitted information into our method.

methods as baselines, and the results demonstrate the robustness of LCA. Our contributions are as follow:

- We build on preliminary experiments to propose insights about bias in evaluating different subjects that further influence the process of uncertainty quantification candidate selecting and analyze the fragility of uncertainty.
- We introduce a novel and flexibility framework for estimating uncertainty, LCA-UQ, which utilizes pointwise KL-Divergence to explicitly account for the discrepancies between candidate answer confidence and the entropy of global samples.
- We evaluate multiple important free-form question-answering datasets in a CoT paradigm on popular pre-trained LLMs ranging from $4B$ to $32B$. Results demonstrate that our LCA-UQ surpass baseline methods. Furthermore, through hyperparameter ablation experiments, we demonstrate how the variables in our method impact the final results and analyze its robustness.

Our data/code is available at https://github.com/linqinhong/LCA.

II. Related Work

Verbalization and logit-based or entropy-based methods play a crucial role in addressing uncertainty in the field of Natural Language Processing (NLP). The verbalization methods which prompt models to output confidence levels for their generated content, first introduced by [13], unfortunately often result in overconfident outputs. Enhancements such as COT reasoning [14] and multi-round dialogue cross models [15] encourage models to stimulate multi-steps reasoning for a more convincing scores. Fine-tuning methods transforms model confidence outputs into assessments of answer correctness in a designed format and tuning the models with specially crafted data [16], [17]. Logit-based and entropy-based methods assess model confidence and uncertainty by focusing on the logits during the output process. Reference [18] add a classification head to the model's final layer, mapping logits to the probability of the "True" token, thus estimating the model's confidence in its responses. Reference [19] combine token-level probabilities and one-sentence entropy to evaluate the uncertainty in model-generated content. Reference [20] proposes to mitigate the miscalibration of token probability caused by linguistic synonymy through data augmentation training and temperature finetuning and [21] suggests that aggregates probabilities of synonymous sentences at the sentence-level in the multi-sampling process for better hallucination detection.

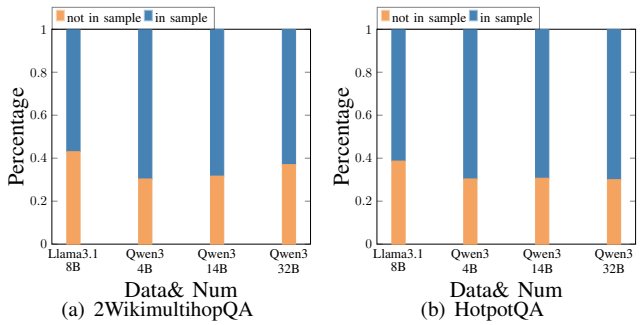
III. Preliminary

Background. Uncertainty Quantification (UQ) aims to compute a prior metric that is highly correlated with answer correctness, used to predict the reliability of the model's output in the absence of a known reference. It can be understood as the entropy of the model's predictions, namely Predictive Entropy (PE). For a given input x and output space Y, the predictive entropy is calculated as follows:

$$PE(x) = -\sum_{y_i \in Y} (P(y_i|x) \log P(y_i|x)) \tag{1}$$

where $P(y_i|x)$ is the conditional probability of a generation y_i. The higher $PE(x)$ is, the closer the model's output probabilities are to a uniform distribution, indicating lower confidence in any specific output y out of the output space Y, and thus greater model uncertainty.

In Bayesian networks, the sampling space for a model with a vocabulary of K tokens generating sequences of length L is exponentially large, specifically $|K|^L$, posing computational challenges. To mitigate these challenges, we can employ Monte



(a) 2WikimultihopQA (b) HotpotQA

Fig. 2: Percentage of 2WikimultihopQA w. & w/o label answers in sample on four models. 'in sample' represents the greedy decoded answer can be entailed by at least one of the sampled answers, while 'not in sample' means it is not entailed by any of the sampled answers.

Carlo sampling [22], which introduces random factors to approximate the sampling process. An unbiased estimate of entropy can be:

$$PE(x) = -\frac{1}{|N|}\sum_{y_i \in Y} \log P(y_i|x) \tag{2}$$

As probabilities tend to decrease with increasing length, length-normalization method [23], replacing probability of y with $\frac{1}{N}\sum_i^N \log P(y_i|y_{<i})$, could be used to scale the conditional probabilities of sentences of different lengths to the same magnitude and has been successfully applied in machine translation scenarios [24].

Bias in Evaluating Different Objects. Previous uncertainty workflows typically involve first sampling a candidate answer through greedy decoding and obtaining probabilities, then sampling N samples to measure entropy in order to estimate the quality of the greedy decoded answer. However, they overlook an important issue: when the number of samples is limited, the sampling space may fail to cover the candidate answer adequately. To analyze the representativeness of the greedy decoded label, we evaluated the relationship between candidate result and the sampled results. We categorize the test question samples based on whether the semantic meaning of the candidate answer is covered by the sample set ("In" vs. "Not in"). Specifically, we first measured the NLI score between the candidate and every samples. Denoting sample set as $\mathcal{S}$ and the greedy decoding answer as g, g is considered to be cover by $\mathcal{S}$ if at least one $NLI(\mathcal{S}_i, \})$ exceeds a predefined threshold α:

$$\text{sim}(\mathcal{S}, \mathcal{G}) = \begin{cases} 1 & if\ \exists\ \text{NLI}(\mathcal{S}_i, \mathcal{G}) > \alpha \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

We set α to 0.5, consistent with prior works. Fig. 2 illustrates the occurrence of greedy decoding results within the sampled outcomes for four models over 2WikimultihopQA [25] and HotpotQA [26]. Our results indicate that in many cases, the candidates do not appear within the sampled set. We further grouped the test data according to whether it is in or not in sample set to analyze the impact on the classification performance of the set entropy. We conduct experiments on these two datasets using three models and two methods. We present the experimental results in Tab. I. In scenarios "In", standard uncertainty metrics exhibit a strong correlation with correctness, yielding high AUROC scores. Conversely, in "Greedy-Sample Mismatch" scenarios ("not in"), we observe a precipitous decline in the discriminative power of the baseline. This phenomenon indicates that the reliability of existing uncertainty workflows is highly contingent on the sampling coverage. Therefore, we emphasize that an uncertainty measurement scheme based on entropy must consider whether the candidate answer is covered by the sampled instances. The simplest yet efficient method is to integrate the candidate into the sample for calculating entropy. In the experiments of this paper, we ensure this condition is strictly met.

TABLE I: Uncertainty estimation AUROC results of *LNPE* & *LUQ-Pair* with and without candidate answers in sample set.

Dataset	Model	LNPE		LUQ-Pair	
		in	not in	in	not in
2WikimultihopQA	Llama-3.1-8B	72.25	39.57	71.52	42.01
	Qwen3-14B	68.80	68.82	76.55	72.68
	Qwen3-4B	63.12	66.11	80.29	50.76
HotpotQA	Llama-3.1-8B	81.76	42.60	85.72	53.47
	Qwen3-14B	64.81	61.59	76.55	72.68
	Qwen3-4B	60.70	40.53	80.29	50.76

Fragility of Uncertainty In the previous section, we provided an example to illustrate the potential fragility when consistency-based and confidence-based methods work in isolation and we conduct a deeper analysis here. Fig. 3 illustrates the relationship between sample-based uncertainty and confidence-based uncertainty for Qwen3-4B [27] on the 2WikimultihopQA dataset, distinguishing between correct and incorrect predictions. For the confidence-based method, we selected TokenSAR [9], while for the entropy-based method, we compute semantic clustering entropy based on TokenSAR. We observe a complex interaction between the two metrics: 1. In low-uncertainty regimes, the model output is consistent, yet the probability score serves as a critical discriminator. Specifically, samples with lower probabilities in this region strongly tend to be incorrect, indicating that consistency without confidence is often a sign of error. 2.In high-uncertainty regimes, despite the dispersion in the semantic space, we observe a significant number of correct answers even with moderate probabilities. These patterns highlight the fragility of relying solely on either uncertainty or probability. The misalignment between high semantic dispersion and local confidence suggests that neither metric alone is sufficient, motivating our proposed fusion approach.

IV. Method

In this section, we present the implementation process of the LCA-UQ framework, which includes: (1) candidate answer

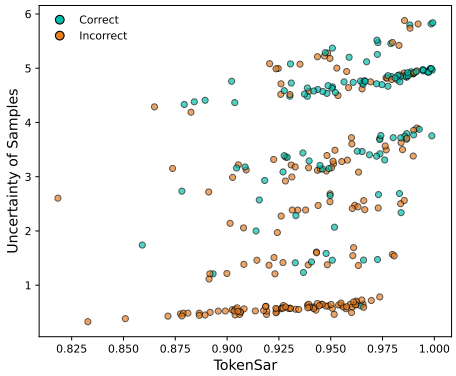


Fig. 3: Visualization of the relationship between sample uncertainty and candidate probability on 2WikimultihopQA. The scattered distribution indicates that relying solely on one dimension is fragile for capturing complex error patterns.

sampling, (2) distribution space estimation, and (3) bridging local calibration and global consistency.

Candidate Answer Sampling. Given a query x and model $\mathcal{M}$ we first generate a candidate answer g via greedy decoding. In order to explore the model's output distribution, we then employed multinomial sampling to generate $|S'|$ sequences, forming the sample set $S' = \{s_1, s_2, ..., s_m\}$. Crucially, to ensure the candidate answer is covered by the semantic space, we construct an augmented set $S = \{s_0, s_1, ..., s_m\}$ where s_0 is the greedy hypothesis g. We then apply a confidence estimation framework to quantify the reliability $C(s_i|x)$ of each prediction. While our method is agnostic to the underlying uncertainty backbone, it can also be calculated from entropy-based metrics such as average token entropy. Let $E(s_i|x)$ denote the measured entropy, we derive the final confidence score $C(s_i|x)$ via the exponential transformation:

$$C(s_i|x) = \exp(-E(s_i|x)). \tag{4}$$

Through $C(g|x)$, we can measure the model's local calibration for the question x. Here, the forced insertion of s_0 into the sample set is a widely used configuration in prior works. Furthermore, as demonstrated in Tab I, this setting is essential.

Distribution space estimation. Through Monte Carlo sampling, we can use the sampled instances to unbiasedly estimate the distribution space predictive entropy of model $\mathcal{M}$ for the question as shown in Eq. 2. In natural language generation tasks, different sentences may express the same meaning, thus sharing a common semantic space. Considering that the correctness of a non-multiple-choice answer should be judged based on its semantics rather than just its lexical content, we calculate the entropy of the sample space after semantic clustering of multiple samples, rather than using the lexical-independent sample entropy. We computed the relevant score $P(entail|s_i, s_j)$ between each answer using a pretrained language model as follow:

$$P(entail|s_i, s_j) = \frac{\exp(l_e)}{\exp(l_e) + \exp(l_c)}, \tag{5}$$

where l_e and l_c are the logits of the "entailment" and "contradiction" classes respectively. We treat samples with a neutral relationship as irrelevant samples as well. Specifically, we used RoBERTa-Large [28] for calculating. Samples with bidirectional semantic entailment will be clustered together, and these samples will ultimately be categorized into $|\mathbb{C}|$ clusters. The conditional probability of a cluster is the sum of the probabilities of the sentences, and we can calculate $E(S|x)$ with Monte Carlo Approximation, the entropy in the semantic space as follows:

$$\begin{aligned} E(S|x) &= -\sum_{c\in\mathbb{C}} C(c_i|x)\log C(c_i|x) \\ &= -\sum_{c\in\mathbb{C}}((\sum_{s_i\in c} C(s_i|x)\log(\sum_{s_i\in c} C(s_i|x)))) \\ &\approx -|\mathbb{C}|^{-1}\sum_{i=1}^{|\mathbb{C}|}\log C(c_i|x). \end{aligned} \tag{6}$$

This entropy encapsulates the global consistency information as well as uncertainty of the model's sampling space. We then calculate the Gibbs probability $C(S|x) = \exp(-E(S|x))$ as the confidence of S, representing the model's perceived probability of a set to be able to provide an answer.

Bridging local calibration and global consistency. We reframe uncertainty estimation as a measurement of divergence between the greedy decoding confidence P (approximated as a point mass) and the aggregated semantic distribution Q derived from sampling. Specifically, to quantify the information loss—or surprisal—incurred when the greedy answer is evaluated against the semantic consensus, we employ the Pointwise Kullback-Leibler divergence (PKL) defined as follow:

$$U(S|x) = P\log\frac{P}{Q} = C(g|x)\cdot\log\frac{C(g|x)}{C(S|x)}. \tag{7}$$

This formulation inherently resolves the limitations identified in our preliminary analysis: 1.confidence Anchoring: the term $C(g|x)$ integrates the model's intrinsic local confidence; 2. mismatch Penalty: as the log-ratio term diverges, it imposes a severe penalty on the uncertainty score.

V. Experiments

Datasets. As the capabilities of LLMs have improved, the scenarios for evaluating UQ methods in short-generation tasks are limited. Therefore, we focus our experimental scenarios on long-generation tasks, CoT specifically. We require models to first generate a chain-of-thought for reasoning and then provide a final answer. We conduct experiments on several free-form text generation tasks in NLP, including two multi-hop question-answering datasets, 2WikimultihopQA [25] and HotpotQA [26], a domain-specific question-answering dataset, MedQA [29], and a mathematical reasoning dataset, Math [30].

Baselines. We selected two classic UQ methods: Length Normalizaiton Predictive Entropy (LNPE) [23], TokenSAR [9], and extended them to long-generation tasks. Additionally, we included three state-of-the-art uncertainty quantification methods work well in long-generation: CoT-UQ [12], LUQ and

TABLE II: Uncertainty estimation AUROCs of our LCA method with different methods as backbone and baselines across datasets. Bold indicates better performance.

model	dataset	LNPE		TokenSAR		Cot-SE		LUQ		LUQ-PAIR	
		Base	LCA	Base	LCA	Base	LCA	Base	LCA	Base	LCA
Llama-3.1-8B	2WikimultihopQA	72.96	**75.04**	72.96	**86.19**	74.79	**87.15**	67.10	**86.37**	64.48	**84.04**
	HotpotQA	77.76	**78.92**	77.76	**86.16**	75.22	**86.97**	82.51	**85.61**	81.18	**85.51**
	MedQA	67.41	**69.74**	67.41	**82.31**	64.98	**82.31**	77.32	**81.43**	76.52	**82.25**
	Math	78.05	**82.01**	78.05	**87.14**	79.67	**87.75**	88.81	**89.60**	**90.37**	89.11
Qwen3-4B	2WikimultihopQA	68.25	**80.15**	68.23	**80.15**	73.37	**81.28**	77.26	**81.80**	77.11	**82.29**
	HotpotQA	67.98	**86.36**	67.99	**86.37**	63.43	**86.64**	83.15	**87.04**	84.11	**86.14**
	MedQA	62.33	**67.20**	62.34	**67.08**	58.81	**66.09**	67.40	**69.73**	65.08	**69.76**
	Math	85.78	**90.57**	85.79	**90.57**	84.69	**91.71**	89.44	**96.58**	81.27	**96.29**
Qwen3-14B	2WikimultihopQA	78.39	**82.11**	78.38	**82.12**	78.81	**82.74**	72.24	**82.85**	73.57	**81.98**
	HotpotQA	67.48	**81.74**	67.47	**81.27**	63.16	**80.35**	81.33	**81.96**	79.66	**83.17**
	MedQA	70.09	**76.70**	70.09	**77.18**	66.14	**74.73**	67.04	**69.91**	65.59	**70.46**
	Math	83.00	**95.06**	83.02	**95.04**	88.01	**95.30**	92.32	**98.76**	90.01	**98.68**
Qwen3-32B	2WikimultihopQA	72.19	**83.48**	72.20	**84.35**	73.23	**84.22**	**82.60**	81.96	**83.35**	79.36
	HotpotQA	68.74	**87.78**	68.75	**87.62**	66.23	**87.31**	84.23	**85.40**	84.20	**84.70**
	MedQA	71.77	**75.69**	71.77	**86.12**	67.77	**85.14**	78.58	**83.99**	78.83	**80.38**
	Math	82.53	**96.74**	82.49	**97.54**	83.18	**97.64**	85.93	**95.82**	87.79	**94.33**
Average		73.42	**81.08**	73.42	**84.83**	72.59	**84.83**	79.83	**84.93**	78.94	**84.28**

LUQ-Pair [11] as the backbone and compared the performance of these methods before and after applying LCA. For CoT-UQ, it is a single-sample confidence evaluation method, and we extend it by combining it with SE. We label it as CoT-SE in the following sections. For other these methods, candidate probabilities are calculated according to the approach outlined in the respective papers. When performing semantic clustering, considering the semantic richness of long-generation answers, directly aggregating them is not sufficiently accurate. Therefore, we disregard CoT and only use the final model output for NLI calculation and classification.

Models. The uncertainty measurement methods we selected require token probabilities. We conducted experiments using popular open-source LLMs, including Llama-3.1-8B [31] and Qwen3-4/14/32B [27], covering models of different sizes.

Metric. For correctness assessment, we leveraged a majority voting ensemble of three closed-source APIs including GPT-4.1-mini[1], DeepSeek-V3.2-Fast[2], and Grok-4-1-Fast-Reasoning[3] to adjudicate response correctness, guaranteeing robust labeling for candidate answers. The reliability of this automated protocol is substantiated in the Appendix, where we demonstrate a high degree of consistency between the LLM voting results and human annotations. For uncertainty method evaluation metric, we adopt Area Under the Receiver Operating Characteristic Curve (AUROC) as the primary metric to assess the efficacy of our uncertainty quantification. This metric serves as a proxy for discriminative power, measuring how reliably the uncertainty scores can separate correct generations from incorrect ones.

Hyperparameters. In the experiments, the top-P and topk settings for model-generated answers were set to the model's default configurations, and diversity in sampling results was ensured through polynomial sampling . For each dataset, we generated five answers per question. All experiments were carried out using two NVIDIA A40 GPUs.

[1]https://openai.com/index/gpt-4-1/

[2]https://chat.deepseek.com/

[3]https://x.ai/grok

VI. Results Analysis

In Tab. II, we provided a detailed performance comparison between the basic baselines and baselines with the LCA-UQ method on evaluation datasets. In the majority of cases, LCA-UQ effectively enhances the performance of the backbone method, demonstrating that PKL successfully links local confidence and global entropy, helping us further distinguish the correctness of answers. The most significant result of LCA-UQ on the test data is observed in the experiment with Llama-3.1-8B on the 2WikimultihopQA dataset, where the AUROC increased from 64.48% to 84.04%, with a 19.56% improvement. On average, LCA-UQ showed the most notable improvement on CoT-SE, with an increase from 72.59% to 84.83%, an improvement of 12.24%. In experiments with Qwen3-14B, even though the LUQ and LUQ-Pair methods achieved AUROCs of 92.32% and 90.01% on the Math dataset, the LCA method can further improve the results to 98.76% and 98.68%, respectively.

VII. Ablation Study

Number of Generation. Fig. 4 reports AUROC versus the number of samples on HotpotQA with Llama-3.1-8B (T=1.0). LCA-UQ methods (solid) consistently outperformcorresponding backbone counterparts (dashed), with the largest gains observed in the low-sample regime (2–5 samples). AUROC increases rapidly as k grows and peaks around 7–12 samples, after which the improvement plateaus or slightly declines, indicating diminishing returns from additional sampling. Task accuracy, shown on the secondary axis, varies more mildly across k, further demonstrating that LCA-UQ effectively enhances the discriminative power of uncertainty estimation methods across different sampling budgets.

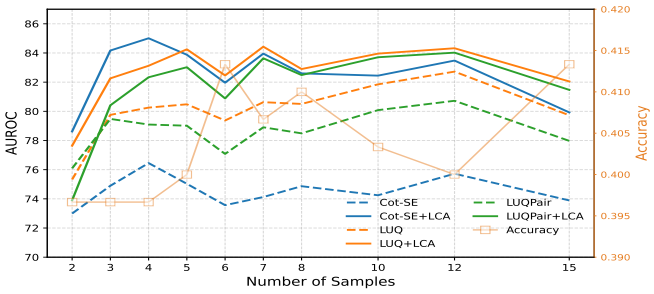


Fig. 4: AUROC versus the number of samples on HotpotQA (Llama-3.1-8B, $T = 1.0$). LCA-enhanced methods (solid) consistently outperform their baselines (dashed), while task accuracy is shown on the secondary axis.

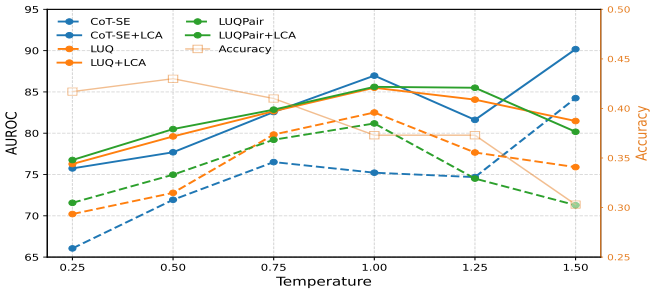


Fig. 5: Effect of temperature on AUROC for uncertainty estimation on HotpotQA. LCA (solid) improves performance over baselines (dashed) across all temperature settings; task accuracy is reported on the right axis.

Temperature. We conducted a temperature ablation experiment using Llama-3.1-8B on HotpotQA to analyze the impact of sampling temperature on AUROC. We show the effect of temperature on performance in Fig. 5. A lower temperature results in sharper token probabilities, which reduces the diversity of the model's generation. As the temperature increases, the diversity and performance gradually improve. However, when the temperature exceeds 1, the generation quality of the LLM begins to decline, and the AUROC of both LUQ and LUQ-Pair decreases. In contrast, CoT-SE shows an increase. We attribute this to the filtering of low-importance words when measuring the entropy of individual samples in the CoT-UQ method. The entropy estimation is less affected by temperature.

Different Integrate Methods One of the core aspects of our work is to quantify uncertainty by effectively measuring the discrepancy between the sampling distribution and the candidate answer distribution. We compare the use of pointswise KL-divergence (PKL) with the trivial method that calculate the mean value of $C(x)$ and $C(S)$ (Mean) and the method Reverse Pointwise KL-divergence (R-PKL) [32] as aggregation methods, where:

$$\mathrm{U}_{R-KLD}(x) = C(S|x) \cdot \log \frac{C(S|x)}{C(g|x)}. \tag{8}$$

As shown in Tab. III, in most cases, the PKL method outperforms R-PKL and Mean. It indicate that when we treat the sampling results as the "correct" distribution and view greedy sampling as the prediction, divergence calculations help us better identify when the model is more likely to be able to answer.

Effectiveness on Multi-fact Generation Multi-fact generation tasks represent a common category within natural language generation (NLG). We took summarization task as a representative. We utilized the Llama-3.1-8B model to conduct experiments on the XSum [33] dataset. Generations with ROUGE-L greater than the threshold will be assigned a label of 1, otherwise it will be assigned a label of 0. The results of these experiments are presented in Tab. IV. Our LCA consistently enhances performance across various methods, achieving a maximum improvement of 0.09 on LNPE backbone. Notably, the method of LNPE performs the best. We attribute this to the presence of multiple facts in the generated text. Specifically, sequence-level clustering employed by other semantic-level methods tends to overlook the independence of individual facts within generations.

VIII. Conclusion

In this paper, we reveal the impact of biases between label sources and samples in uncertainty estimation and propose our LCA method to aggregate the confidence of them. Results demonstrate that our method surpasses the state-of-the-art performance. Further ablation results show the impact of various parameters on method performance.

IX. Limitations

We recognize that there are several areas where our approach can be further enhanced: (1) Model Capability: Employing a more powerful model, or fine-tuning Roberta on the test domain to assess semantic relevance could yield superior sampling results for semantic clustering and would significantly boost the performance of our uncertainty measurement. (2) Similarity Calculation in Multi-Fact Scenarios: Experiments on the Xsum dataset demonstrate that poor sequence-level similarity calculations can undermine the performance in multi-fact contexts. Implementing more refined similarity calculations in these scenarios is likely to improve overall model performance.

Acknowledgement

This work was supported in part by Research Project of Quan Cheng Laboratory, China (Grant No.QCL20250203) and in part by the National Natural Science Foundation of China under Grant 62172053 and Grant 62302059.

References

[1] M. Fomicheva, S. Sun, L. Yankovskaya, F. Blain, F. Guzmán, M. Fishel, N. Aletras, V. Chaudhary, and L. Specia, "Unsupervised quality estimation for neural machine translation," *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 539–555, 2020.

[2] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

TABLE III: The performance of PKL-based, Mean-Based, and R-PKL-based method with three backbone methods on two datasets using three models.

model	dataset	CoT-SE				LUQ				LUQ-Pair			
		Base	Mean	PKL	R-PKL	Base	ADD	PKL	R-PKL	Base	Mean	PKL	R-PKL
Llama-3.1-8B	2Wiki	74.79	**87.13**	87.15	82.73	67.10	75.76	**84.04**	69.63	64.48	73.01	**86.37**	67.10
	HotpotQA	75.22	79.73	**86.97**	64.07	82.51	85.85	**85.51**	73.63	81.18	83.21	**85.61**	69.41
Qwen3-4B	2Wiki	78.81	82.71	**82.74**	82.51	72.24	78.50	**81.98**	69.57	73.57	79.77	**82.85**	71.97
	HotpotQA	63.16	79.57	80.35	**81.21**	81.33	**83.49**	83.17	74.72	79.66	**81.98**	81.96	73.82
Qwen3-14B	2Wiki	73.37	81.08	**81.28**	80.53	77.26	80.02	**82.29**	70.58	77.11	80.57	**81.80**	71.49
	HotpotQA	63.43	86.14	**86.64**	86.13	83.15	**86.74**	86.14	81.16	84.11	87.02	**87.04**	82.77

TABLE IV: Results of Llama3.1-8B on Xsum under different Rouge Threshold.

Thres	**LNPE**		**TokenSAR**	
	base	LCA	base	LCA
0.3	0.543	**0.614**	0.529	**0.557**
0.2	0.525	**0.616**	0.500	**0.532**
0.15	0.548	**0.636**	0.518	**0.552**

[3] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, "Llama: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.

[4] Q. Dong, L. Li, D. Dai, C. Zheng, Z. Wu, B. Chang, X. Sun, J. Xu, and Z. Sui, "A survey on in-context learning," *arXiv preprint arXiv:2301.00234*, 2022.

[5] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou *et al.*, "Chain-of-thought prompting elicits reasoning in large language models," *Advances in neural information processing systems*, vol. 35, pp. 24 824–24 837, 2022.

[6] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei, "Scaling laws for neural language models," *arXiv preprint arXiv:2001.08361*, 2020.

[7] H. Luo, Q. Sun, C. Xu, P. Zhao, J. Lou, C. Tao, X. Geng, Q. Lin, S. Chen, and D. Zhang, "Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct," *arXiv preprint arXiv:2308.09583*, 2023.

[8] L. Kuhn, Y. Gal, and S. Farquhar, "Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation," *arXiv preprint arXiv:2302.09664*, 2023.

[9] J. Duan, H. Cheng, S. Wang, C. Wang, A. Zavalny, R. Xu, B. Kailkhura, and K. Xu, "Shifting attention to relevance: Towards the uncertainty estimation of large language models," *arXiv preprint arXiv:2307.01379*, 2023.

[10] Y. F. Bakman, D. N. Yaldiz, B. Buyukates, C. Tao, D. Dimitriadis, and S. Avestimehr, "Mars: Meaning-aware response scoring for uncertainty estimation in generative llms," *arXiv preprint arXiv:2402.11756*, 2024.

[11] C. Zhang, F. Liu, M. Basaldella, and N. Collier, "Luq: Long-text uncertainty quantification for llms," *arXiv preprint arXiv:2403.20279*, 2024.

[12] B. Zhang and R. Zhang, "Cot-uq: Improving response-wise uncertainty quantification in llms with chain-of-thought," *arXiv preprint arXiv:2502.17214*, 2025.

[13] S. Lin, J. Hilton, and O. Evans, "Teaching models to express their uncertainty in words," *arXiv preprint arXiv:2205.14334*, 2022.

[14] M. Xiong, Z. Hu, X. Lu, Y. Li, J. Fu, J. He, and B. Hooi, "Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms," *arXiv preprint arXiv:2306.13063*, 2023.

[15] R. Cohen, M. Hamri, M. Geva, and A. Globerson, "Lm vs lm: Detecting factual errors via cross examination," *arXiv preprint arXiv:2305.13281*, 2023.

[16] S. Kapoor, N. Gruver, M. Roberts, A. Pal, S. Dooley, M. Goldblum, and A. Wilson, "Calibration-tuning: Teaching large language models to know what they don't know," in *Proceedings of the 1st Workshop on Uncertainty-Aware NLP (UncertaiNLP 2024)*, 2024, pp. 1–14.

[17] H. Han, T. Li, S. Chen, J. Shi, C. Du, Y. Xiao, J. Liang, and X. Lin, "Enhancing confidence expression in large language models through learning from past experience," *arXiv preprint arXiv:2404.10315*, 2024.

[18] S. Kadavath, T. Conerly, A. Askell, T. Henighan, D. Drain, E. Perez, N. Schiefer, Z. Hatfield-Dodds, N. DasSarma, E. Tran-Johnson *et al.*, "Language models (mostly) know what they know," *arXiv preprint arXiv:2207.05221*, 2022.

[19] Y. Huang, J. Song, Z. Wang, S. Zhao, H. Chen, F. Juefei-Xu, and L. Ma, "Look before you leap: An exploratory study of uncertainty measurement for large language models," *arXiv preprint arXiv:2307.10236*, 2023.

[20] Z. Jiang, J. Araki, H. Ding, and G. 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*, vol. 9, pp. 962–977, 2021.

[21] S. Farquhar, J. Kossen, L. Kuhn, and Y. Gal, "Detecting hallucinations in large language models using semantic entropy," *Nature*, vol. 630, no. 8017, pp. 625–630, 2024.

[22] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *international conference on machine learning*. PMLR, 2016, pp. 1050–1059.

[23] A. Malinin and M. Gales, "Uncertainty estimation in autoregressive structured prediction," *arXiv preprint arXiv:2002.07650*, 2020.

[24] K. Murray and D. Chiang, "Correcting length bias in neural machine translation," *arXiv preprint arXiv:1808.10006*, 2018.

[25] X. Ho, A.-K. D. Nguyen, S. Sugawara, and A. Aizawa, "Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps," *arXiv preprint arXiv:2011.01060*, 2020.

[26] Z. Yang, P. Qi, S. Zhang, Y. Bengio, W. Cohen, R. Salakhutdinov, and C. D. Manning, "Hotpotqa: A dataset for diverse, explainable multi-hop question answering," in *Proceedings of the 2018 conference on empirical methods in natural language processing*, 2018, pp. 2369–2380.

[27] A. Yang, A. Li, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Gao, C. Huang, C. Lv *et al.*, "Qwen3 technical report," *arXiv preprint arXiv:2505.09388*, 2025.

[28] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[29] D. Jin, E. Pan, N. Oufattole, W.-H. Weng, H. Fang, and P. Szolovits, "What disease does this patient have? a large-scale open domain question answering dataset from medical exams," *Applied Sciences*, vol. 11, no. 14, p. 6421, 2021.

[30] D. Hendrycks, C. Burns, S. Kadavath, A. Arora, S. Basart, E. Tang, D. Song, and J. Steinhardt, "Measuring mathematical problem solving with the math dataset," *arXiv preprint arXiv:2103.03874*, 2021.

[31] A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Yang, A. Fan *et al.*, "The llama 3 herd of models," *arXiv e-prints*, pp. arXiv–2407, 2024.

[32] A. Malinin and M. Gales, "Reverse kl-divergence training of prior networks: Improved uncertainty and adversarial robustness," *Advances in neural information processing systems*, vol. 32, 2019.

[33] S. Narayan, S. B. Cohen, and M. Lapata, "Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization," *arXiv preprint arXiv:1808.08745*, 2018.

TABLE V: Result with selecting the answer with max likelihood inside max semanticn cluster.

model	dataset	LNPE		TokenSAR		CoT-SE		LUQ		LUQ-Pair	
		Base	LCA	Base	LCA	Base	LCA	Base	LCA	Base	LCA
Llama-3.1-8B-Instruct	2Wiki	72.96	75.04	72.96	86.19	74.79	87.15	67.10	84.04	64.48	86.37
	hotpotqa	77.76	76.92	77.76	86.16	75.22	86.97	82.51	85.51	81.18	85.61
	MedQA	67.41	59.74	67.41	82.31	64.98	82.31	77.32	82.25	76.52	81.43
	math	78.05	82.01	78.05	87.14	79.67	87.75	88.81	89.11	90.37	89.60
Qwen3-4B	2Wiki	68.16	79.58	68.13	79.59	72.86	80.68	77.15	81.40	77.06	81.23
	hotpotqa	67.86	86.07	67.86	86.08	63.38	86.44	83.23	86.13	84.19	87.02
	MedQA	61.84	66.52	61.84	66.40	58.79	65.72	67.57	69.82	65.10	69.78
	math	85.78	90.57	85.79	90.57	84.69	91.71	89.44	96.29	81.27	96.58
Qwen3-14B	2Wiki	78.29	82.57	78.29	82.57	78.36	83.05	72.11	82.18	73.50	83.03
	hotpotqa	67.48	81.74	67.47	81.27	63.16	80.35	81.33	83.17	79.66	81.96
	MedQA	70.23	76.67	70.23	77.32	66.38	74.86	67.00	70.57	66.12	70.18
	math	83.00	95.06	83.02	95.04	88.01	95.30	92.32	98.68	90.01	98.76
Qwen3-32B	2Wiki	72.04	83.03	72.05	83.91	73.47	83.86	82.83	79.24	83.59	81.78
	hotpotqa	68.74	87.78	68.75	87.62	66.23	87.31	84.23	84.70	84.20	85.40
	MedQA	71.77	75.69	71.77	86.12	67.77	85.14	78.58	80.38	78.83	83.99
	math	82.53	96.74	82.49	97.54	83.18	97.64	85.93	94.33	87.79	95.82

APPENDIX

A. Human Annotation Study

To assess the reliability of our LLM-based judging protocol, we perform a small-scale human verification study on outputs from two representative models (LLaMA-3.1-8B and Qwen3-4B) across three datasets (Math, HotpotQA, and TriviaQA). For each model–dataset combination, we randomly select 200 instances and manually evaluate one sampled response per instance. Human annotators provide binary correctness labels based solely on the semantic validity of the final answer. We then measure the agreement between human judgments and the automatic labels produced by our judge ensemble. As shown in Tab. VI, the consistency remains high across all settings, indicating that the LLM-as-judge labeling is generally reliable.

TABLE VI: Human annotation subsets by model size and dataset.

Model	**Dataset**	**Consistency**
LLama3.1-8B	Math	98%
	HotpotQA	98%
	TriviaQA	99%
Qwen3-4B	Math	99%
	HotpotQA	98%
	TriviaQA	99%

B. Different Candidate Ablation

Consider a scenario where we sample 5 samples. Although the candidate answer has the highest conditional probability, it is not the one with the highest conditional probability in the largest semantic cluster in the semantic space. To produce a more reliable answer, we should consider candidates from the largest semantic cluster as our selection criterion. We have implemented this approach in all the results in the main experiment. The experimental results in Tab. V demonstrate that our method still maintains strong robustness and stability.