Towards a Unified View of Uncertainty Estimation for Large Language Model with Internal State

Anonymous ACL submission

Abstract

Recent advancements have seen a dramatic increase in the use of large language models (LLMs) due to their impressive text generation capabilities. However, these models often produce confident yet inaccurate predictions, highlighting the critical need for effective uncertainty assessment. Various approaches to estimate uncertainty have emerged, primarily utilizing the token probabilities in model predictions, yet the relationship and distinctions among these methods warrant deeper exploration. This study investigates the essential design elements of current uncertainty estimation techniques and introduces a comprehensive framework for evaluating uncertainty in LLMs. Our findings reveal that uncertainty information is dispersed across tokens, and the model's certainty about its uncertainty strengthens after making a prediction. Additionally, we present a novel, efficient supervised method, Adaptive Uncertainty Probing (AUP), which significantly surpasses previous methods in performance. Through extensive experiments across multiple models on five distinct datasets, we validate the effectiveness, generalizability, and efficiency of our method.

1 Introduction

Large Language Models (LLMs) have demonstrated remarkable capabilities across various natural language processing (NLP) tasks, including question answering, summarization, and conversation generation (Touvron et al., 2023; Zeng et al., 2022; Singhal et al., 2022; Brown et al., 2020). Despite their success, LLMs often face reliability issues such as hallucinations and factual inaccuracies (Hu et al., 2023; Amayuelas et al., 2023; Yin et al., 2023). These challenges are particularly concerning because users might unknowingly trust incorrect information generated by LLMs due to their confident and convincing language output. Therefore, accurately estimating the confidence of the model's predictions becomes crucial. Uncertainty estimation aims to generate a value indicating the confidence level of an LLM's prediction, which helps filter out potentially harmful information before it reaches the user, thus mitigating the impact of hallucinations.

The estimation of uncertainty has been widely studied before the emergence of large language models. Following Gal, the uncertainty of a neural network is typically classified into *aleatoric uncertainty* and *epistemic uncertainty*. *Aleatoric uncertainty* typically arises from noise present in the training data, which can stem from inaccuracies or outdated information. Conversely, *epistemic uncertainty* is attributed to variations in a model's parameters, as a multitude of potential models could explain a particular training dataset. This paper concentrates on the estimation of *predictive uncertainty*, which combines both *aleatoric uncertainty* and *epistemic uncertainty* and represents the model's confidence in a prediction.

The estimation of uncertainty has been extensively studied (Gal and Ghahramani; Gal; Malinin et al., 2017; Lahlou et al., 2023). Several heuristic and empirical methods have been proposed to estimate the uncertainty of an LLM's predictions. These methods primarily rely on reasonable priors derived from observations of LLM behavior and can generally be categorized into single-inference and multi-inference methods. Single-inference methods operate on the assumption that the probability of a token serves as a partial indicator of the model's uncertainty. However, using token probability for uncertainty estimation has drawbacks, such as overconfidence (Zhang et al., 2023), where the model assigns higher probabilities to words that have been previously mentioned in the text. Another issue is the generation inequality problem (Duan et al., 2023), where irrelevant token probability values can disrupt accurate estimation of sentence-level uncertainty. In contrast,

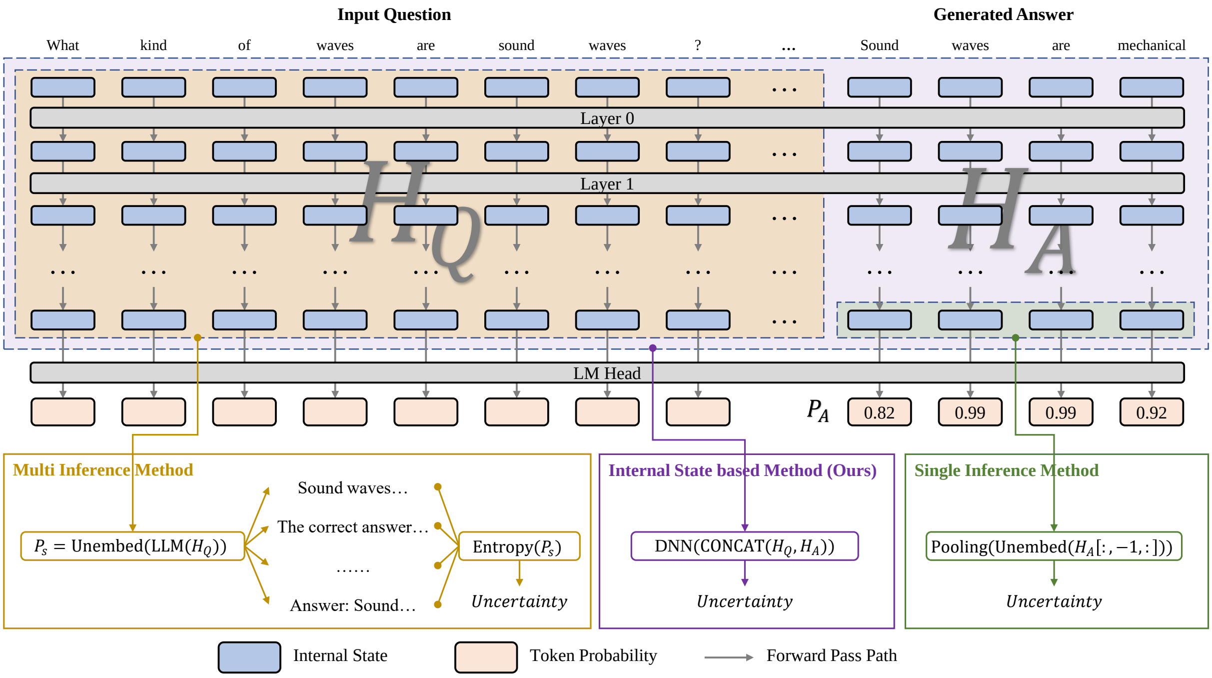


Figure 1: Illustration of the unified framework and various types of uncertainty estimation methods.

multi-inference methods (Lin et al., 2022b; Kuhn et al., 2023) suggest that the entropy of the generation space increases with the model's uncertainty. However, these methods incur high computational costs as they require sampling multiple responses and constructing a similarity matrix from these responses. Despite the variety of methods available, the relationship between different categories of uncertainty estimation methods has not been thoroughly investigated.

In this paper, we will focus on the following three critial questions in uncertainty estimation of LLM. Firstly, we examine the comparative effectiveness and efficiency of single-inference versus multi-inference methods, detailed in Section 3.1. Secondly, we address the challenges and limitations associated with using token probability as a measure of uncertainty, as these methods often fail to capture true uncertainty accurately. Lastly, we identify and discuss the key design elements that differentiate the most effective methods for estimating uncertainty, elaborating on how these elements enhance the accuracy and reliability of the estimates in Section 3.2.

To answer these questions, we conduct extensive experiments and develop several variants of internal state-based uncertainty estimation methods. We introduce a novel method, Adaptive Uncertainty Probing (AUP), which incorporates the optimal design choices identified in our analysis. Our method significantly outperforms existing techniques in terms of effectiveness, generalization, and efficiency.

Our contributions can be summarized as follows:

- **Unified Framework for Uncertainty Estimation:** We introduce a comprehensive framework that systematically organizes existing uncertainty estimation methods in LLMs into two main categories: single-inference and multi-inference approaches. This framework clarifies the differences and linkages between the methods, enhancing our understanding of their foundational principles and enabling a clearer comparison and evaluation. This organization aims to advance the field of uncertainty estimation by providing a structured base for further research and development.
- **Experimental Insights:** Utilizing our unified framework, we conduct extensive experiments to examine key design choices within various uncertainty estimation methods. Our findings affirm that uncertainty information remains consistent across different token positions, and that the model's internal state is critical for accurate uncertainty prediction. These insights highlight the need for a detailed analysis of model internals and underscore the importance of empirical validation in refining design choices to enhance method effective-

ness.

- **Adaptive Uncertainty Probing:** Building on the insights from our framework and experimental findings, we propose Adaptive Uncertainty Probing (AUP), a new lightweight supervised method optimized for uncertainty estimation. AUP is evaluated against several benchmark methods across diverse datasets, showing superior performance in terms of effectiveness, generalization, and efficiency. AUP exemplifies how theoretical advancements can translate into practical applications, offering a significant improvement over existing methods.

2 Related Work

Following Huang et al. (2023), we categorize existing methods for uncertainty estimation in LLM into two main types: single-inference and multi-inference methods.

Single-Inference Methods: Single-inference methods utilize the token probability distribution information of a single LLM prediction to estimate the uncertainty of the sentence. Malinin and Gales (2021) explore uncertainty estimation in auto-regressive structured prediction tasks using a unified and interpretable probabilistic ensemble-based framework. Duan et al. (2023) introduce the SAR series techniques, which simultaneously assess the importance of individual tokens and sentences, and readjust attention during uncertainty estimation. These approaches use the token probability of the model's prediction to gauge uncertainty in an unsupervised fashion. However, the token probability lacks calibration with uncertainty, resulting in sub-optimal performance.

Multi-Inference Methods: Multi-inference methods leverage the stochasticity of the model in generating text by perturbing the model's generation slightly and then estimating uncertainty using the divergence among those predictions. Lin et al. (2022b) introduce Lexical Similarity, which calculates uncertainty by the similarity of the generated response strings. Kuhn et al. (2023) introduce Semantic Entropy, which utilizes a bi-directional entailment algorithm to cluster semantically equivalent samples for better uncertainty estimation. Manakul et al. (2023) introduce SelfCheckGPT, a sampling-based method for hallucination detection. Zhang et al. (2024) introduce LUQ, a new approach for assessing sentence-level consistency in lengthy textual contexts. Gao et al. (2024) propose a new perturbation sampling-based uncertainty quantification framework designed specifically for LLMs. These methods still operate under the same assumption as single-inference methods. Additionally, the process of sampling multiple answers may pose computational challenges. Li et al. (2024) introduce a novel self-detection paradigm that considers the comprehensive answer space beyond LLM-generated answers.

Other Related Works: Recently, there emerge some internal state-based uncertainty estimation methods.Azaria and Mitchell (2023) first discovered that the internal state of large language models can be used for true and false judgment. Xiong et al. (2023) perform a comparative assessment of verbalized-based and consistency-based techniques for uncertainty estimation. Lin et al. (2022a) discovered that the fine-tuned GPT-3 model can provide confidence in its responses using natural language. Kadavath et al. (2022) explore fine-tuning a large language model with an extra value head to determine which questions the models can answer. Ahdritz et al. (2024) explore detecting epistemic uncertainty by employing a more extensive language model for guidance, which focuses on separating epistemic uncertainty from aleatoric uncertainty. Liang et al. (2024) introduce the Knowledge State Probing method. Initially, it explores estimating uncertainty within model internal states, but it only uses the internal states of the question when estimating uncertainty. Kossen et al. (2024) proposed SEPs that directly estimate SE from the hidden states within a single generation. Chen et al. (2024) propose EigenScore to better evaluate responses' self-consistency. However, these works employ simple mean pooling or last pooling for uncertainty aggregation. As a result, they are unable to model the significance of different internal states across the layer and tokens. **Main difference of AUP:** Our proposed method, Adaptive Uncertainty Probing (AUP), distinguishes itself from existing methods by focusing on the internal states of LLMs to estimate uncertainty. Unlike single-inference methods that rely on token probability, which often leads to overconfidence and multi-inference methods that are computationally expensive, AUP leverages the optimal design choices identified through our unified framework and empirical validation. Specifically, AUP uses a lightweight supervised approach to probe the internal states of LLMs, capturing both epistemic and aleatoric uncertainty without

Method	Information Source	Probing Function	Pooling Function
	Probability-Based		
Single Inference Method	$H_A[-1]$	probability-based	sum/mean/...
Multi Inference Method	H_Q	probability-based	entropy
	Internal State-Based		
Knowledge State Probing	H_Q	internal state-based	last
Adaptive Knowledge State Probing	H_Q	internal state-based	adaptive
Last Uncertainty Probing	H_A	internal state-based	last
Mean Uncertainty Probing	H_A	internal state-based	mean
Adaptive Uncertainty Probing	H_A	internal state-based	adaptive

Table 1: We examine the key design choices of different uncertainty estimation methods by comparing existing methods across various categories and proposing some variants.

the need for multiple sampling or perturbations. Compared to existing internal state-based methods, AUP models the importance of activations from different tokens to probing uncertainty information through an Adaptive Pooling Mechanism. This approach not only improves the accuracy of uncertainty estimation but also enhances the efficiency and generalization across various datasets. Our method's ability to integrate insights from different design elements into a coherent and effective uncertainty estimation technique marks a significant advancement over previous methods.

3 Bridging the Gap - A Unified View

3.1 A Unified Framework for single and multi inference methods

Given an L layer large language model with the hidden size of D and vocabulary V, a tokenized question $Q \in \mathbb{R}^{N_Q}$, a tokenized answer A to be evaluated, and a maximum new tokens N_A for generation. * Eq. 1 shows the LLM's generation space, which comprises a total of $|V|^{N_A}$ possible unique predictions. Each prediction consists of N_A tokens, collectively forming a set S.

$$S = \{(x_1, ..., x_{N_A}) \mid x_i \in V, 1 \leq i \leq N_A\}. \quad (1)$$

We utilize the LLM operator to describe the operation of obtaining the internal state H_X from a given text input X, as shown in Eq. 2. When each prediction in set S is combined with Q and applying the LLM operator, we get the hidden state matrix H_S, which contains the internal state of all potential predictions, as shown in Eq. 4.

$$\mathrm{LLM}(X) = \mathrm{Stack}(\{h_l \mid 0 \leq l \leq L\}). \quad (2)$$

$$H_S = \mathrm{LLM}(\mathrm{Concat}(Q, S)), \quad (3)$$

$$H_S \in \mathbb{R}^{|V|^{N_A} \times N_A \times L \times D}. \quad (4)$$

Subsequently, the probability matrix P_S is obtained with the $\mathrm{Unembed}$ operator, which maps the final layer of H_S with $\mathrm{Head_{LM}}$ and applies the $\mathrm{Softmax}$ function, denoted in Eq. 5.

$$P_S = \mathrm{Unembed}(H_S), P_S \in \mathbb{R}^{|V|^{N_A} \times N_A}. \quad (5)$$

The element $p_{j,k}$ in probability matrix P_S is the token probability of the k-th token in the j-th possible prediction, which is an extremely high-dimensional sparse matrix, as most of the predictions have a generative probability close to zero.

The connection and difference: The pooling and entropy functions of predictive entropy are listed in Eq. 6:

$$U_{single} = -\sum_{j}^{N_A} \log(P_S[i_A]_j) \quad (6)$$

$$U_{multi} = -\sum_{i}^{|V|^{N_A}} \sum_{j}^{N_A} \log(P_{S\,i,j}) \prod_{j}^{N_A} P_{S\,i,j} \quad (7)$$

Upon examining Eq. 6 and Eq. 7, it becomes evident that both single and multiple inference techniques operate under a common assumption: **a higher token probability suggests reduced uncertainty.** The variance lies in how these methods approach this notion. Single inference methods focus on the absolute generative probability P_A, whereas multi-inference methods factor in the relative relationship between P_A and P_S.

*Without loss of generality, we assume that $A \in \mathbb{R}^{N_A}$. For answers shorter than N_A tokens in length, we can pad the answer with special tokens (e.g., EOS) that have a token probability of one.

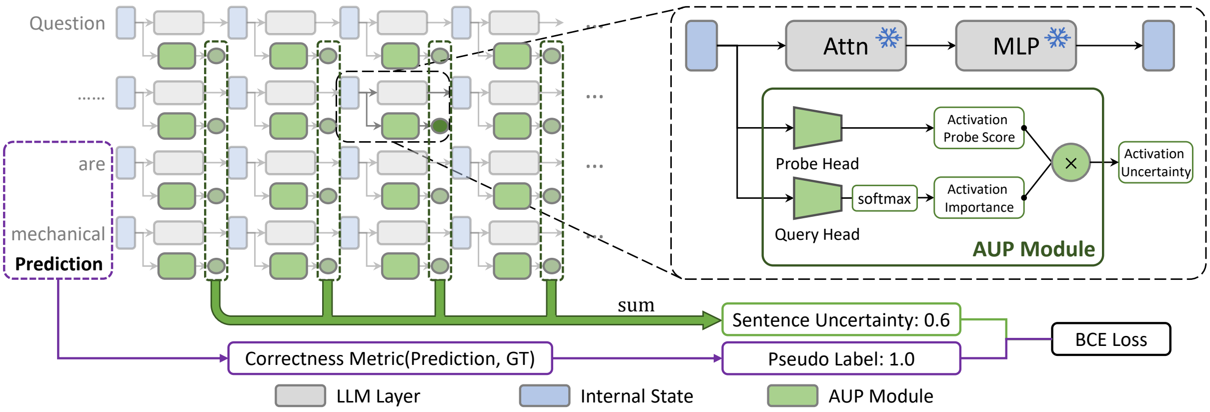


Figure 2: The proposed Adaptive Uncertainty Probing method.

3.2 Key design choice of existing methods

SelfCheckGPT (Manakul et al., 2023) highlights a positive correlation between token probability and factuality in the LLM's prediction. Nonetheless, the token probability is not flawless as it may not consistently signify the model's certainty across all situations. A recent study (Zhang et al., 2023) reveals that the language model may exhibit excessive confidence when the preceding context includes superficial tokens that appear to be associated with an imagined token, or when the context itself is affected by exposure bias because of the auto-regressive characteristic of the generative process. Another study (Duan et al., 2023) reveals that the probability of irrelevant tokens contributes little to the uncertainty of the sentence. **While token probability occasionally touches on uncertainty, its primary connection lies with text generation, encompassing a nuanced concept beyond solely reflecting uncertainty.** Therefore, a more robust and accurate feature is required to assess the uncertainty of a sentence due to the insufficient calibration of probability about uncertainty.

We summarize the key design choices of different methods in Table 1 and propose four variants. As shown in the table, We decompose the uncertainty estimation method into three defined design dimensions: The "Information Source" describes the specific method's reliance on certain sources of information to estimate uncertainty. The "Probing Function" delineates the form of a function utilized by a specific method to extract uncertainty from the information source. The "Pooling Function" specifies the form of function employed by a specific method to aggregate token-level uncertainty into sentence-level uncertainty. Building on this decomposition, we introduce our adaptive uncertainty probing method, which we will elaborate on in the subsequent section and the experiment section.

4 Adaptive Uncertainty Probing

Following the findings of the preceding section, token probability emerges as a feature derived via the UnEmbedding operator through an unsupervised method. However, despite its derivation, token probability lacks proper calibration with uncertainty. Consequently, there arises a necessity to extract a more refined feature from the internal states of the model.

Motivated by these insights, we introduce Adaptive Uncertainty Probing (AUP), a lightweight supervised approach. AUP utilizes probing heads to extract uncertainty signals from the model's internal states. Additionally, we integrate an adaptive uncertainty pooling mechanism to detect uncertain tokens, thereby improving sentence-level uncertainty estimation. **Direct Uncertainty Probing:** As shown in Figure 2, given a tokenized question $Q \in \mathbb{R}^{H_Q}$ and answer $A \in \mathbb{R}^{H_A}$, our initial step involves employing the LLM operator to extract the hidden state of the answer. N_Q and N_A are the length of the question and answer. We denote The internal states of Q and A in Eq. 10:

$$H_{QA} = \mathrm{LLM}(\mathrm{Concat}(Q, A)) \tag{8}$$

$$H_Q = H[:N_Q] \in \mathbb{R}^{N_Q \times L \times D} \tag{9}$$

$$H_A = H[N_Q:] \in \mathbb{R}^{N_A \times L \times D}. \tag{10}$$

Next, as shown in Figure 2, we present trainable probing heads after every LLM layer that are structured as a 2-layer MLP with a moderately narrow hidden size in between. The probing heads take the hidden states of the model's response H_A as input and assess the uncertainty score for each in-

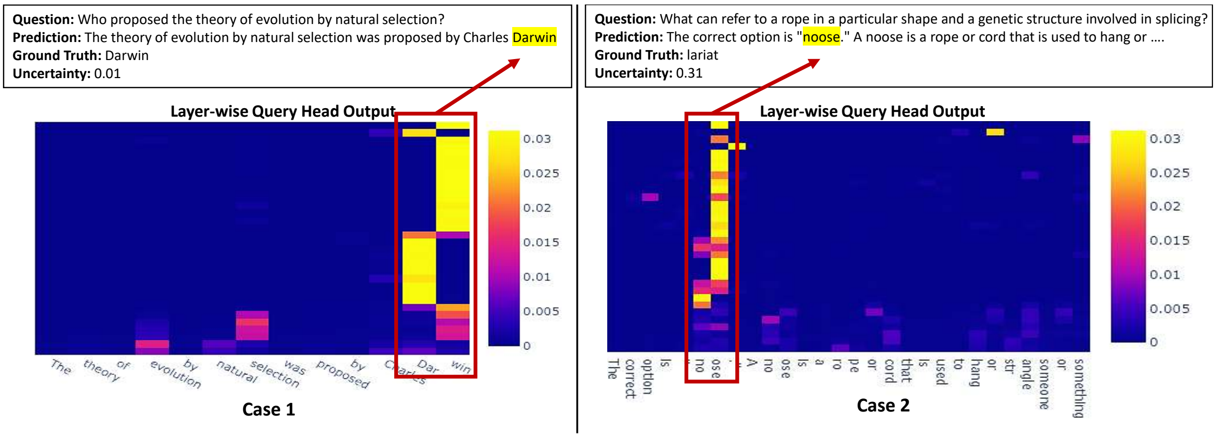


Figure 3: Case study of the proposed Adaptive Uncertainty Probing method, we visualize the output of the query head. Our method successfully identifies key tokens and assigns them high weight to improve the estimation of sentence-level uncertainty

ternal state across all layers and tokens $H_{i,j}$, as represented in Eq. 11:

$$Score_A = \text{Head}_{\text{prob}}(H_A) \tag{11}$$

Adaptive Uncertainty Pooling: The uncertainty within a sentence can span various layers and token positions. The model's confidence in generating the last token does not necessarily indicate confidence in generating intermediate positions, and vice versa. The self-attention mechanism may not always aggregate the uncertainty information of the entire sentence. Thus, Instead of relying on the self-attention mechanism of the LLM to *aggregate* uncertainty information, we have devised an adaptive uncertainty pooling mechanism to enhance sentence-level uncertainty pooling, just as we utilize the probing layer instead of the Unembedding operator to *extract* uncertainty information from the hidden state.

As shown in Eq. 12, for each layer, a query head (which is also structured as a 2-layer MLP) is incorporated to assess the significance of each activation, and then we apply a softmax function to obtain $Weight_A$, the weight of all activation. This allows for a more precise assessment of uncertainty across different layers and token positions.

$$Weight_A = \text{Softmax}(\text{Head}_{\text{query}}(H_A) \tag{12}$$

Finally, we aggregate the scores of all activations to obtain the ultimate uncertainty for the entire sentence, which is shown in Eq. 13.

$$Uncertainty = Weight_A \cdot Score_A \tag{13}$$

We also present a case study of the proposed Adaptive Uncertainty Probing (AUP) method to illustrate its capability in pinpointing key tokens, which are defined as those crucial for estimating the sentence's uncertainty. For instance, in one example, "Charles Darwin" are identified as key tokens within a prediction; in another, it is "noose". We then apply AUP to two sample questions and visualize the outputs from the query head, demonstrating how the model identifies which tokens to focus on when estimating sentence-level uncertainty. As shown in Figure 3, the query head consistently targets the key token regardless of its position in the sentence, showcasing a robust generalization capability. Importantly, the model achieved this without specific supervision on the location of key tokens, learning to identify them autonomously.

Training with pseudo label: As shown in Figure 2, we use the existing training splits from various question-answering datasets like SciQ, TriviaQA, CoQA, MedMCQA, and MedQA. We first get the correctness value of the model's prediction by calculating the correctness metric such as Include, Rouge-L and SentSim. Then we use the correctness value directly as the pseudo uncertainty label for the training of the AUP module. The LLM's parameters are frozen and are used only to generate the internal state, only the AUP module is trained using binary cross-entropy loss.

5 Empirical Evaluation

5.1 Experiment Setup

Baselines: We compare our approach with various baselines. For single-inference techniques, we consider Predictive Entropy(PE) (Kadavath et al., 2022), Length-Normalized Predictive Entropy(LN-

Model	Dataset	Single Inference Methods PE	LN-PE	TokenSAR	Multi Inference Methods SentSAR	SAR	LS	SE	Internal State Methods KSP	AUP(Ours)
Vicuna-7B	SciQ	66.31	62.70	60.29	66.51	60.91	59.05	62.26	65.04	**87.50**
	CoQA	63.43	65.09	47.99	64.13	54.13	60.69	63.93	65.16	**76.70**
	TriviaQA	66.30	59.00	57.85	66.31	62.57	65.92	61.54	68.01	**75.99**
	MedMCQA	62.84	65.59	59.07	62.34	60.96	61.09	51.11	65.90	**77.04**
	MedQA	62.08	59.62	53.01	61.96	57.08	62.69	54.40	70.92	**81.06**
	Average	64.19	62.40	55.64	64.25	59.13	61.89	58.65	67.00	**79.66**
Vicuna-13B	SciQ	69.27	63.55	59.51	69.96	61.08	60.53	69.60	73.65	**84.55**
	CoQA	64.03	66.25	50.42	65.09	53.05	59.16	64.91	73.79	**80.25**
	TriviaQA	74.26	67.83	65.85	74.79	71.02	70.90	74.17	69.64	**77.10**
	MedMCQA	63.15	62.61	59.86	63.52	59.70	59.33	60.86	68.70	**76.12**
	MedQA	58.57	55.34	51.16	59.14	53.66	55.92	58.20	64.76	**70.49**
	Average	65.86	63.12	57.36	66.50	59.70	61.17	65.55	70.11	**77.70**
Llama3-8B-IT	SciQ	70.22	62.91	62.46	71.21	65.76	61.51	69.47	81.64	**82.84**
	CoQA	67.84	58.50	45.63	66.77	56.58	57.72	65.57	71.33	**73.61**
	TriviaQA	62.46	56.75	48.38	62.42	50.64	55.27	60.59	66.67	**71.25**
	MedMCQA	67.97	61.57	62.02	67.83	66.49	65.15	67.22	70.88	**73.49**
	MedQA	60.87	57.36	49.59	60.55	52.86	56.06	59.70	65.52	**67.92**
	Average	65.87	59.42	53.62	65.76	58.47	59.14	64.51	71.21	**73.82**

Table 2: **Main Results:** We compare eight representative methods from various categories on the validation set of five datasets. We evaluate the correctness of the answers using the metric *include* and assess the performance of each method using AUROC. Detailed results with different correctness metrics can be found in the appendix E.

PE) (Malinin and Gales, 2021), and TokenSAR (Duan et al., 2023). Regarding multi-inference methods, we analyze SentSAR, Lexical Similarity(LS) (Lin et al., 2022b), and Semantic Entropy(SE) (Kuhn et al., 2023). Additionally, we incorporate Knowledge State Probing (KSP) (Liang et al., 2024) as the current state-of-the-art internal state-based method.

Datasets and Models: We study five diverse datasets: (1) SciQ with 13.7K science exam questions (Welbl et al., 2017); (2) CoQA, tailored for Conversational Question Answering (Reddy et al., 2019); (3) TriviaQA, featuring over 650K question-answer-evidence triples (Joshi et al., 2017); (4) MedMCQA, a large-scale multi-subject medical question dataset (Pal et al., 2022); (5) MedQA, sourced from medical exams (Jin et al., 2020). For each dataset, we use 1000 samples from the validation split for evaluation. We conduct experiments on the Vicuna-7B, Vicuna-13B and Llama-3-8B-IT models to verify the generalization of our method.

Correctness Metric: We employ a rule-based template matching approach called *include* as our primary correctness metric. We also incorporate other commonly used metrics for question answering: Rouge-L (Lin, 2004) and SentSim (Reimers and Gurevych, 2019). Detailed results with different correctness metrics can be found in the appendix G.

Evaluation Metric: Following prior work (Kuhn et al., 2023), we assess uncertainty by framing it as the task of determining whether to trust a model's prediction for a given prompt. We measure this using the AUROC metric, which indicates the probability that a correct answer has a lower uncertainty score than an incorrect one. Higher AUROC scores signify better uncertainty estimation.

Implementation Details: We generally followed the setup in (Kuhn et al., 2023). For each question, we generate up to 256 tokens and we sample 10 answers for multi-inference methods. For internal state-based methods, we use 2000 samples from the training split of the corresponding datasets and 1000 samples from the validation split for testing. We train models on a single A100 GPU for 10 epochs using Adam optimizer with a learning rate of 1e-4 and a batch size of 16. The prompt template and more implementation details are included in appendix I

5.2 Main Results

We assess eight different methods representing various categories on the validation set of five datasets. The accuracy of the responses is measured using the *include* metric and the performance of each method is evaluated based on AUROC. We include more results of other correctness metrics (Rouge-L) in appendix E.

Model	Dataset	Ours	Pooling Ablation			Source Ablation
		AUP	Mean Pooling	Last Pooling	Last Pooling Large	Question Only
Vicuna-7B	SciQ	87.21±0.21	74.25±0.01	83.57±0.06	83.91±0.77	64.86±0.42
	CoQA	76.88±0.25	73.84±0.25	73.83±0.19	71.98±1.79	62.98±0.22
	TriviaQA	75.56±0.21	73.98±0.27	75.10±0.05	75.17±0.58	68.33±0.07
	MedMCQA	77.08±0.47	73.20±0.12	74.59±0.04	74.69±0.84	65.16±0.21
	MedQA	80.82±0.03	76.16±0.00	74.82±0.15	74.57±1.11	65.80±1.43
	Average	**79.51±0.23**	74.29±0.13	76.38±0.10	76.06±1.02	65.43±0.47
Vicuna-13B	SciQ	84.71±0.57	76.45±0.00	79.79±0.13	79.13±0.42	71.56±0.00
	CoQA	79.95±0.00	77.14±0.00	77.31±0.24	76.79±0.32	59.02±16.41
	TriviaQA	76.74±0.13	75.47±0.23	75.70±0.25	75.62±0.14	69.13±0.29
	MedMCQA	75.98±0.20	72.69±0.00	73.84±0.40	71.80±1.26	66.82±0.51
	MedQA	69.68±0.85	66.81±0.03	65.25±0.17	64.07±0.71	60.91±0.23
	Average	**77.41±0.35**	73.71±0.05	74.38±0.24	73.48±0.57	65.49±3.49

Table 3: **Ablation Results:** The results of the ablation study indicate a significant improvement in model performance when the adaptive pooling mechanism is included in the architecture.

As shown in the table, our method outperforms the baseline approaches across all datasets by a large margin. Our method also surpasses Knowledge State Probing (KSP), the current leading internal state-based method. This superiority may stem from two potential reasons: 1) KSP relies on the hidden state of the input question, neglecting the valuable information present in the model's answer. 2) KSP exclusively utilizes the hidden state of the last token, which may lack comprehensive uncertainty information about the entire sentence.

5.3 Ablation Study

To evaluate the efficacy of our method, we do ablation study focusing on the influence of various components on the model's overall performance. This included a comparison between Adaptive Uncertainty Probing and its four derivatives. Regarding the adaptive pooling mechanism, we tested three alternatives: Mean Pooling, which computes sentence-level uncertainty using the average score across the sentence; Last Pooling, which bases uncertainty assessment exclusively on the final token; and Last Pooling Large, an augmented version of Last Pooling that incorporates an increased parameter count to assess the effect of model capacity on performance. For the information source ablation, we investigated the Question Only variant, which utilizes only the question's internal state. The results clearly demonstrate that our model outperforms all variants, affirming the effectiveness of integrating the adaptive pooling mechanism into our framework. Further, the ablation study offers crucial insights:

Distribution of Uncertainty Information Across Tokens: The results from the pooling ablation experiments suggest that neither using a mean function, focusing solely on the last token, nor simply enlarging the model parameters led to performance improvements. This underscores the efficacy of the adaptive pooling mechanism in effectively aggregating uncertainty information across different tokens, thereby underlining the necessity for more sophisticated strategies to enhance model performance.

Enhanced Certainty about Uncertainty Post-Prediction: The findings from the source ablation reveal that while leveraging information solely from the question yields some benefit, it does not match the effectiveness of utilizing the internal state derived from the answer in the standard configuration. This points to an increase in the model's certainty about its uncertainty following the prediction, suggesting a more nuanced understanding of uncertainty as the model processes the answer.

6 Conclusion

We develop a comprehensive unified framework that integrates existing methods for uncertainty estimation in LLMs. Through extensive experiments and analysis, we reveal that uncertainty information is preserved across different token positions and that the internal state of the answer is particularly valuable for predicting uncertainty. We propose a novel, lightweight supervised method named Adaptive Uncertainty Probing, and demonstrate that it significantly outperforms existing methods in terms of effectiveness, generalization, and efficiency.

Limitations

The Adaptive Uncertainty Probing method isn't directly applicable to uncertainty estimation in closed-source models. However, applying our method to closed-source models is possible using open-source proxy models, which we see as a promising research direction. Additionally, dataset bias may affect the generalization performance of our method. A promising research direction involves preventing the probing head from learning basic heuristic knowledge.

References

Gustaf Ahdritz, Tian Qin, Nikhil Vyas, Boaz Barak, and Benjamin L. Edelman. 2024. Distinguishing the Knowable from the Unknowable with Language Models. ArXiv:2402.03563 [cs] version: 1.

Alfonso Amayuelas, Liangming Pan, Wenhu Chen, and William Wang. 2023. Knowledge of Knowledge: Exploring Known-Unknowns Uncertainty with Large Language Models. ArXiv:2305.13712 [cs].

Amos Azaria and Tom Mitchell. 2023. The Internal State of an LLM Knows When It's Lying. ArXiv:2304.13734 [cs].

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. ArXiv:2005.14165 [cs].

Chao Chen, Kai Liu, Ze Chen, Yi Gu, Yue Wu, Mingyuan Tao, Zhihang Fu, and Jieping Ye. 2024. INSIDE: LLMs' Internal States Retain the Power of Hallucination Detection. ArXiv:2402.03744 [cs].

Jinhao Duan, Hao Cheng, Shiqi Wang, Alex Zavalny, Chenan Wang, Renjing Xu, Bhavya Kailkhura, and Kaidi Xu. 2023. Shifting Attention to Relevance: Towards the Uncertainty Estimation of Large Language Models. ArXiv:2307.01379 [cs].

Yarin Gal. Uncertainty in Deep Learning.

Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning.

Xiang Gao, Jiaxin Zhang, Lalla Mouatadid, and Kamalika Das. 2024. SPUQ: Perturbation-Based Uncertainty Quantification for Large Language Models. ArXiv:2403.02509 [cs].

Mengting Hu, Zhen Zhang, Shiwan Zhao, Minlie Huang, and Bingzhe Wu. 2023. Uncertainty in Natural Language Processing: Sources, Quantification, and Applications. ArXiv:2306.04459 [cs].

Yuheng Huang, Jiayang Song, Zhijie Wang, Shengming Zhao, Huaming Chen, Felix Juefei-Xu, and Lei Ma. 2023. Look Before You Leap: An Exploratory Study of Uncertainty Measurement for Large Language Models. ArXiv:2307.10236 [cs].

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. 2020. What Disease does this Patient Have? A Large-scale Open Domain Question Answering Dataset from Medical Exams. ArXiv:2009.13081 [cs].

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. 2022. Language Models (Mostly) Know What They Know. ArXiv:2207.05221 [cs].

Jannik Kossen, Jiatong Han, Muhammed Razzak, Lisa Schut, Shreshth Malik, and Yarin Gal. 2024. Semantic Entropy Probes: Robust and Cheap Hallucination Detection in LLMs. ArXiv:2406.15927.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic Uncertainty: Linguistic Invariances for Uncertainty Estimation in Natural Language Generation. ArXiv:2302.09664 [cs].

Salem Lahlou, Moksh Jain, Hadi Nekoei, Victor Ion Butoi, Paul Bertin, Jarrid Rector-Brooks, Maksym Korablyov, and Yoshua Bengio. 2023. DEUP: Direct Epistemic Uncertainty Prediction. ArXiv:2102.08501 [cs, stat].

Moxin Li, Wenjie Wang, Fuli Feng, Fengbin Zhu, Qifan Wang, and Tat-Seng Chua. 2024. Think Twice Before Trusting: Self-Detection for Large Language Models through Comprehensive Answer Reflection. ArXiv:2403.09972.

Yuxin Liang, Zhuoyang Song, Hao Wang, and Jiaxing Zhang. 2024. Learning to Trust Your Feelings: Leveraging Self-awareness in LLMs for Hallucination Mitigation. ArXiv:2401.15449 [cs].

Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022a. Teaching Models to Express Their Uncertainty in Words. ArXiv:2205.14334 [cs].

Zi Lin, Jeremiah Zhe Liu, and Jingbo Shang. 2022b. Towards Collaborative Neural-Symbolic Graph Semantic Parsing via Uncertainty. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 4160–4173, Dublin, Ireland. Association for Computational Linguistics.

Andrey Malinin and Mark Gales. 2021. Uncertainty Estimation in Autoregressive Structured Prediction. ArXiv:2002.07650 [cs, stat].

Andrey Malinin, Anton Ragni, Kate Knill, and Mark Gales. 2017. Incorporating Uncertainty into Deep Learning for Spoken Language Assessment. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 45–50, Vancouver, Canada. Association for Computational Linguistics.

Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. 2023. SelfCheckGPT: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models. ArXiv:2303.08896 [cs].

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. 2022. MedMCQA : A Large-scale Multi-Subject Multi-Choice Dataset for Medical domain Question Answering. ArXiv:2203.14371 [cs].

Siva Reddy, Danqi Chen, and Christopher D. Manning. 2019. CoQA: A Conversational Question Answering Challenge. ArXiv:1808.07042 [cs].

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. ArXiv:1908.10084 [cs].

Karan Singhal, Shekoofeh Azizi, Tao Tu, S. Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, Perry Payne, Martin Seneviratne, Paul Gamble, Chris Kelly, Nathaneal Scharli, Aakanksha Chowdhery, Philip Mansfield, Blaise Aguera y Arcas, Dale Webster, Greg S. Corrado, Yossi Matias, Katherine Chou, Juraj Gottweis, Nenad Tomasev, Yun Liu, Alvin Rajkomar, Joelle Barral, Christopher Semturs, Alan Karthikesalingam, and Vivek Natarajan. 2022. Large Language Models Encode Clinical Knowledge. ArXiv:2212.13138 [cs].

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. LLaMA: Open and Efficient Foundation Language Models. ArXiv:2302.13971 [cs] version: 1.

Johannes Welbl, Nelson F. Liu, and Matt Gardner. 2017. Crowdsourcing Multiple Choice Science Questions. *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 94–106. Conference Name: Proceedings of the 3rd Workshop on Noisy User-generated Text Place: Copenhagen, Denmark Publisher: Association for Computational Linguistics.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2023. Can LLMs Express Their Uncertainty? An Empirical Evaluation of Confidence Elicitation in LLMs. ArXiv:2306.13063 [cs].

Zhangyue Yin, Qiushi Sun, Qipeng Guo, Jiawen Wu, Xipeng Qiu, and Xuanjing Huang. 2023. Do Large Language Models Know What They Don't Know? ArXiv:2305.18153 [cs].

Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, Weng Lam Tam, Zixuan Ma, Yufei Xue, Jidong Zhai, Wenguang Chen, Peng Zhang, Yuxiao Dong, and Jie Tang. 2022. GLM-130B: An Open Bilingual Pre-trained Model. ArXiv:2210.02414 [cs].

Caiqi Zhang, Fangyu Liu, Marco Basaldella, and Nigel Collier. 2024. LUQ: Long-text Uncertainty Quantification for LLMs. ArXiv:2403.20279 [cs].

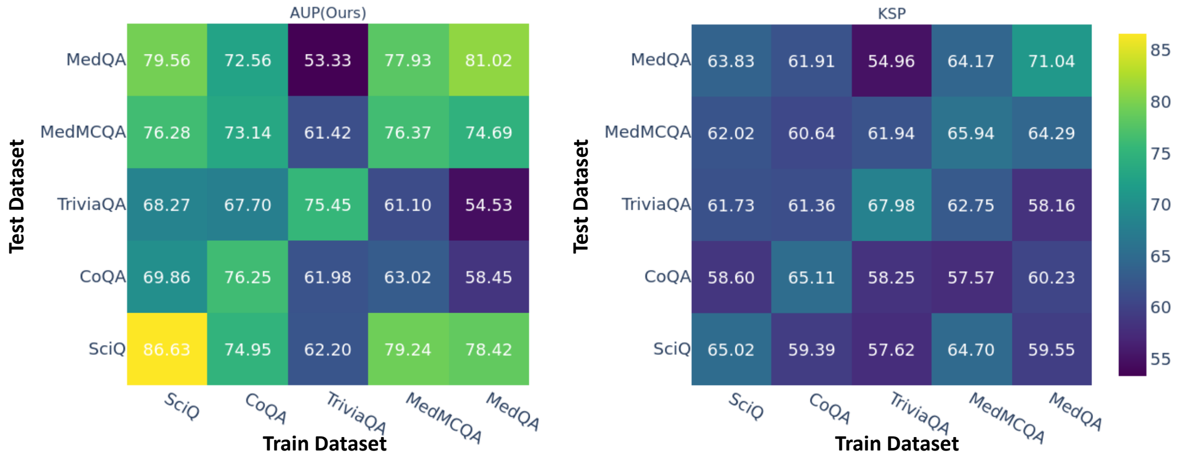
Tianhang Zhang, Lin Qiu, Qipeng Guo, Cheng Deng, Yue Zhang, Zheng Zhang, Chenghu Zhou, Xinbing Wang, and Luoyi Fu. 2023. Enhancing Uncertainty-Based Hallucination Detection with Stronger Focus. ArXiv:2311.13230 [cs].

A Out-of-Domain Generalization

We assess the out-of-domain generalization capability of the proposed approach using cross-validation. Our model is trained on a specific dataset and then tested on a different dataset. We compare our method with KSP. The results are demonstrated in Figure 4. As indicated in the table, our approach demonstrates superior out-of-domain generalization capability when compared to the KSP. Notably, our method's out-of-domain performance remains better than non-internal state-based methods.

B Qualitative Analysis

We also present a case study of the proposed Adaptive Uncertainty Probing (AUP) method to illustrate its capability in pinpointing key tokens, which are defined as those crucial for estimating the sentence's uncertainty. For instance, in one example, "Charles Darwin" are identified as key tokens within a prediction; in another, it is "noose". We then apply AUP to two sample questions and visualize the outputs from the query head, demonstrating



	AUP(Ours)	KSP
Average ID Performance	**79.15**	67.02
Average OOD Performance	**68.43**	60.68

Figure 4: **Out-of-Domain Generalization Results:** We assess the OOD performance of internal state-based methods on five datasets. AUP represents our approach and KSP refers to Knowledge State Probing.

how the model identifies which tokens to focus on when estimating sentence-level uncertainty. As shown in Figure 3, the query head consistently targets the key token regardless of its position in the sentence, showcasing a robust generalization capability. Importantly, the model achieved this without specific supervision on the location of key tokens, learning to identify them autonomously.

C Sensitivity Analysis

In addition to the ablation study, we further investigate the influence of various hyper-parameters within the AUP framework. This exploration specifically focuses on the hidden size of the MLP in the probing head and the scale of training data. Our analysis involves multiple iterations on the SciQ dataset employing the Vicuna-7B model, under diverse seeding conditions. As depicted in Figure 5, AUP demonstrates robustness against fluctuations in the MLP's hidden size, maintaining stable performance throughout these variations. Furthermore, augmenting the volume of training data enhances the efficacy of AUP, although the incremental benefits diminish as the dataset size increases. These findings indicate an improved learning capacity and enhanced generalization, affirming AUP's adaptability across training datasets of varying magnitudes.

Method	Training	Inference		Total
		Sampling	Calculation	
PE(LN-PE)	-	-	0.07s	0.07s
TokenSAR	-	-	18.83s	18.83s
SentSAR	-	47.80s	5.72s	53.52s
LS	-	47.80s	7.02s	54.82s
SE	-	47.80s	778.51s	826.3s
KSP	8.23min	-	28.13s	28.13s
AUP(Ours)	8.53min	-	28.13s	28.13s

Table 4: The extra computational expenses incurred by various uncertainty estimation techniques.

D Computation Efficiency

We also evaluate the computational efficiency of various uncertainty estimation methods, including our AUP framework, by examining the additional computational overhead these methods introduce. This assessment was carried out by performing 1000 inference runs on the SciQ dataset, where for methods requiring multiple inferences per prediction, such as ensemble techniques, we sampled 10 predictions to estimate uncertainty. This analysis not only complements our understanding of AUP's performance scalability but also highlights its computational practicality in operational environments.

As shown in Table 4, the additional computational load imposed by various uncertainty estimation methods can be divided into two main categories: the direct computation of uncertainty and the overhead linked to generating multiple predictions (relevant for multi-inference methods). The

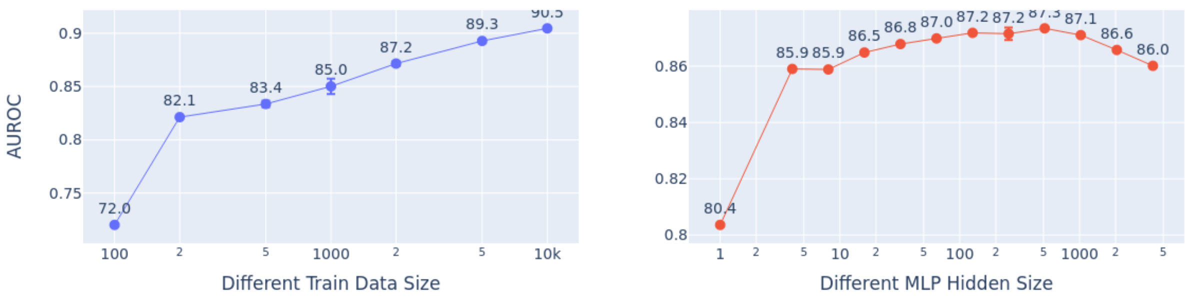


Figure 5: Sensitivity analysis of training data size and MLP hidden size, revealing interesting trends in the performance of the AUP when changing these hyper-parameters.

Model	Dataset	Single Inference Methods			Multi Inference Methods				Internal State Methods	
		PE	LN-PE	TokenSAR	SentSAR	SAR	LS	SE	KSP	AUP(Ours)
Vicuna-7B	SciQ	68.29	60.50	61.00	67.35	61.98	59.66	62.51	66.79	**87.73**
	CoQA	73.15	68.74	51.44	72.26	52.63	67.31	70.18	76.73	**86.41**
	TriviaQA	82.59	54.45	53.56	81.78	67.08	76.15	77.73	75.70	**86.69**
	MedMCQA	62.92	59.56	55.81	63.55	52.37	54.52	46.90	73.52	**84.36**
	MedQA	67.74	63.99	55.86	65.62	56.29	61.62	47.58	70.41	**83.47**
	Average	70.94	61.45	55.53	70.11	58.07	63.85	60.98	72.63	**85.73**
Vicuna-13B	SciQ	77.57	62.30	60.62	77.03	62.16	63.87	75.73	70.39	**88.68**
	CoQA	76.07	74.13	58.10	76.20	50.14	68.23	74.70	80.60	**88.04**
	TriviaQA	81.03	69.26	67.89	81.21	74.76	76.71	81.15	72.95	**82.67**
	MedMCQA	67.44	62.22	59.51	67.45	60.64	62.92	63.56	73.90	**84.15**
	MedQA	69.98	62.47	55.89	68.99	55.93	60.04	64.87	72.65	**80.90**
	Average	74.42	66.08	60.40	74.17	60.72	66.36	72.00	74.10	**84.89**

Table 5: Main Results using Rouge-L as the correctness metric: The correctness threshold is adjusted individually for each dataset to maintain the model's accuracy consistent.

AUP method distinguishes itself by its significantly reduced additional computational overhead compared to other techniques. This efficiency stems from its ability to compute the uncertainty score directly during the inference phase without the necessity for sampling multiple answers.

E Main Results with Different Correctness Metrics

We conduct a comparison of our method against seven other baseline models using two additional correctness metrics: Rouge-L and SentSim. As shown in Table 5 and Table 6, our method consistently outperforms the baselines across all evaluated metrics, thereby demonstrating its superior performance under various correctness criteria.

F More Experiments with Llama3-8B-Instruct

We conducted more experiments with the newer and more popular Llama3-8B-Instruct backbone. As show in Table 7, our approach maintains a strong performance advantage on newer LLM backbones. This demonstrates the excellent generalization of the approach.

G Correctness Metric

The primary correctness metric, denoted as *include*, is defined as follows: the model's prediction should contain only the correct answer in the predicted text. In datasets without answer choices (such as CoQA and TriviaQA), a prediction is deemed correct if the correct answer is found in the first sentence of the prediction. Conversely, in datasets with answer choices (such as SciQ, MedMCQA, and MedQA), a prediction is considered accurate if the correct answer is present in the first sentence of the prediction, and any incorrect choices are not included in that first sentence. This rule-based metric has proven to be simple and effective when an appropriate prompt is given, as it eliminates the need for manually setting a judgment threshold as required by Rouge-L and SentSim.

For Rouge-L and SentSim, we utilize the F1 score of these metrics as the measure of correctness. When calculating the AUROC metric during evaluation, a correctness threshold is required to

Model	Dataset	Single Inference Methods			Multi Inference Methods				Internal State Methods	
		PE	LN-PE	TokenSAR	SentSAR	SAR	LS	SE	KSP	AUP(Ours)
Vicuna-7B	SciQ	64.18	59.41	56.00	64.35	57.15	57.11	58.50	75.73	**87.03**
	CoQA	61.01	60.83	51.08	61.21	51.50	58.49	61.01	79.62	**85.64**
	TriviaQA	82.32	53.29	53.27	82.44	66.49	75.55	78.45	76.89	**87.59**
	MedMCQA	60.85	59.86	55.05	60.93	53.27	53.12	47.61	69.16	**81.18**
	MedQA	65.30	63.79	55.48	64.68	56.79	60.08	51.59	69.69	**82.48**
	Average	66.73	59.44	54.18	66.72	57.04	60.87	59.43	74.22	**84.78**
Vicuna-13B	SciQ	64.06	56.54	55.27	64.53	56.19	56.04	63.67	78.77	**86.10**
	CoQA	65.18	68.24	56.00	65.77	49.84	62.51	65.68	84.81	**89.16**
	TriviaQA	81.02	68.51	66.93	81.31	74.87	76.19	81.15	76.15	**83.74**
	MedMCQA	65.03	61.42	58.67	66.04	58.08	57.81	60.56	70.26	**79.68**
	MedQA	64.16	58.43	51.49	64.15	54.95	59.03	61.73	67.47	**75.58**
	Average	67.89	62.63	57.67	68.36	58.79	62.31	66.56	75.49	**82.85**

Table 6: Main Results using SentSim as the correctness metric: The correctness threshold is adjusted individually for each dataset to maintain the model's accuracy consistent.

Correctness Metric	Dataset	Single Inference Methods			Multi Inference Methods				Internal State Methods	
		PE	LN-PE	TokenSAR	SentSAR	SAR	LS	SE	KSP	AUP(Ours)
Include	SciQ	70.22	62.91	62.46	71.21	65.76	61.51	69.47	81.64	82.84
	CoQA	67.84	58.50	45.63	66.77	56.58	57.72	65.57	71.33	73.61
	TriviaQA	62.46	56.75	48.38	62.42	50.64	55.27	60.59	66.67	71.25
	MedMCQA	67.97	61.57	62.02	67.83	66.49	65.15	67.22	70.88	73.49
	MedQA	60.87	57.36	49.59	60.55	52.86	56.06	59.70	65.52	67.92
	Average	65.87	59.42	53.62	65.76	58.47	59.14	64.51	71.21	73.82
Rouge-L	SciQ	75.79	47.55	40.30	74.15	55.08	63.90	73.03	81.02	93.31
	CoQA	75.00	57.37	53.35	74.32	49.38	66.14	73.30	73.51	84.03
	TriviaQA	58.27	43.86	45.51	58.87	51.73	51.25	55.73	69.07	78.59
	MedMCQA	71.54	65.01	62.11	71.08	68.65	68.94	69.29	73.03	75.77
	MedQA	63.09	60.34	50.91	62.83	53.60	59.42	61.47	68.19	69.23
	Average	68.74	54.83	50.44	68.25	55.69	61.93	66.56	72.96	80.19
SentSim	SciQ	74.25	61.27	59.15	75.10	67.82	65.03	73.07	77.72	79.11
	CoQA	70.61	53.37	47.44	69.68	52.81	63.59	68.02	79.37	87.07
	TriviaQA	56.14	41.07	45.19	56.47	51.84	48.15	52.86	74.99	86.72
	MedMCQA	68.11	62.16	61.34	68.49	67.67	67.24	66.81	72.53	75.30
	MedQA	63.82	61.16	49.62	63.97	54.11	59.93	62.33	69.78	72.71
	Average	66.59	55.81	52.55	66.74	58.85	60.79	64.62	74.88	80.18

Table 7: Main Results with Llama3-8B-Instruct. The correctness threshold of the Rouge-L and SentSim is adjusted individually for each dataset to maintain the model's accuracy consistent.

categorize predictions into true and false classes. Following the primary correctness metric, we set the correctness threshold separately to keep the accuracy under this correctness threshold coherent with the accuracy when using the primary correctness metric. We provided the correctness threshold of each dataset in the Table.

H Sensitivity to Correctness Threshold

We investigate the sensitivity of our methods to correctness threshold. Figure 6 and Figure show the effects of applying different thresholds of correctness metrics. Higher thresholds mean stricter correctness standards. We also visualize the accuracy curve under different thresholds. As the metrics become stricter, the performances of uncertainty quantization are affected. However, our methods consistently outperform baseline methods. 8

I Prompt Template and Data Samples

We utilize the standard system prompt from the Vicuna series models to structure the question within the training and testing dataset. For the SciQ, MedMCQA, CoQA, and MedQA datasets, we shuffle the options randomly and organize the questions with a fixed template; For TriviaQA, we use a 10-shot prompt. To streamline predictions, only the first sentence of the model's output is retained. In Figure 8, we present a few examples from our

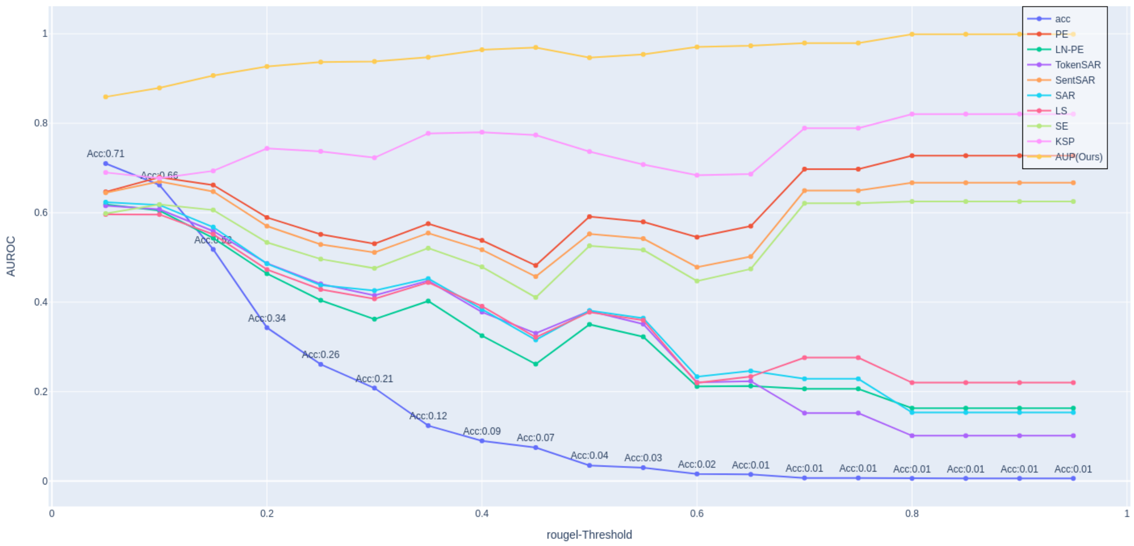


Figure 6: The performance of different methods over various Rouge-L thresholds. Results are obtained from the Vicuna-7B model on the SciQ dataset.

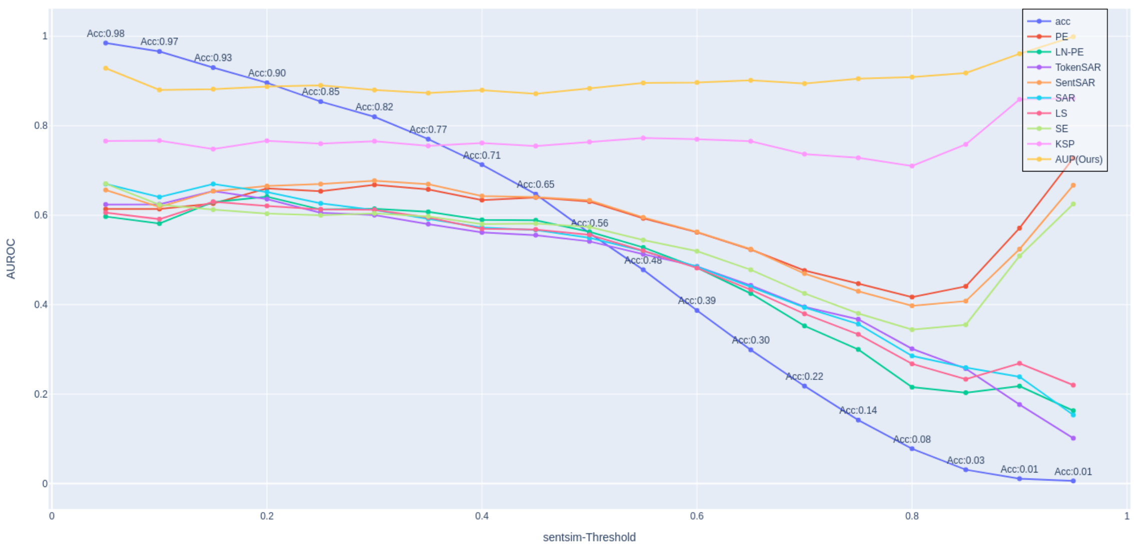


Figure 7: The performance of different methods over various Sentence Similarity thresholds. Results are obtained from the Vicuna-7B model on the SciQ dataset.

Dataset: SciQ

Prompt:
A chat between a curious user and an artificial intelligence assistant. The assistant gives helpful, detailed, and polite answers to the user's questions. USER:Question:How many cycles do cells have? Options:two, four, six, seven Answer: ASSISTANT:

Prediction(Vicuna-7B):
Cells have one cycle.

Dataset: CoQA

Prompt:
A chat between a curious user and an artificial intelligence assistant. The assistant gives helpful, detailed, and polite answers to the user's questions. USER:Context:One day Poppy the puppy went outside to play. He ran around the yard as fast as he could. After running, Poppy began to play with his ball. When Poppy was tired of that, Poppy play catch with Mary. Poppy was very happy when Mary wanted to play with him. Poppy would jump on Mary and start to lick her face. Mary was happy when Poppy showed her all that love. Later on Mary and Poppy went inside to eat and take a nap. Question:Who is Poppy? Answer: ASSISTANT:

Prediction(Vicuna-7B):
Poppy is a puppy who went outside to play.

Dataset: TriviaQA

Prompt:
A chat between a curious user and an artificial intelligence assistant. The assistant gives helpful, detailed, and polite answers to the user's questions. USER:Question:Do You Know Where You're Going To? was the theme from which film? Answer:mahogany Question:Which actor had a Doberman Pinscher called Kirk? Answer:william shatner Question:Which musical featured the song Thank Heaven for Little Girls? Answer:gigi Question:The VS-300 was a type of what? Answer:helicopter Question:Where in England was Dame Judi Dench born? Answer:york Question:Who won Super Bowl XX? Answer:chicago bears Question:Which American-born Sinclair won the Nobel Prize for Literature in 1930? Answer:sinclair lewis Question:Stapleton international airport is in which US state? Answer:colorado Question:How is musician William Lee Conley better known? Answer:william lee conley broonzy, big bill broonzy, bill broonzy Question:Who wrote the novel Evening Class? Answer:maeve binchy Question:In which state is Camp David? Answer: ASSISTANT:

Prediction(Vicuna-7B):
Maryland.

Dataset: MedMCQA

Prompt:
A chat between a curious user and an artificial intelligence assistant. The assistant gives helpful, detailed, and polite answers to the user's questions. USER:Question:The intermediate host for T. Saginata is: Options:Man, Cow, Dog, Pig Answer: ASSISTANT:

Prediction(Vicuna-7B):
The intermediate host for T.

Dataset: MedQA

Prompt:
A chat between a curious user and an artificial intelligence assistant. The assistant gives helpful, detailed, and polite answers to the user's questions. USER:Question:A group of scientists is verifying previous research on DNA replication. In the picture is the theoretical structure for tRNA. Where is the binding site for an amino acid? Options:A, B, D, C Answer: ASSISTANT:

Prediction(Vicuna-7B):
The binding site for an amino acid on tRNA is located in the anticodon loop region.

Figure 8: Data Samples of the testing datasets

Model	Dataset	Test Accuracy	$\text{Threshold}_{RougeL}$	$\text{Threshold}_{SentSim}$
Vicuna-7B	SciQ	66.1	0.11	0.44
	CoQA	62.6	0.17	0.41
	TriviaQA	35.7	0.40	0.80
	MedMCQA	21.8	0.24	0.54
	MedQA	26.6	0.17	0.54
Vicuna-13B	SciQ	71.5	0.12	0.44
	CoQA	64.8	0.12	0.40
	TriviaQA	40.5	0.50	0.86
	MedMCQA	30.8	0.22	0.51
	MedQA	31.6	0.18	0.58
Llama3-8B	SciQ	92.0	0.33	0.64
	CoQA	76.4	0.20	0.47
	TriviaQA	46.7	0.22	0.59
	MedMCQA	56.0	0.50	0.76
	MedQA	58.6	0.25	0.63

Table 8: Test Accuracy and Correctness Threshold of different correctness metrics for different datasets.

dataset.