Query-Level Uncertainty in Large Language Models

Lihu Chen[1], **Gerard de Melo**[2], **Fabian M. Suchanek**[3], **Gaël Varoquaux**[4]
[1]Imperial College London, United Kingdom
[2]Hasso Plattner Institute / University of Potsdam, Potsdam, Germany
[3]Telecom Paris, Institut Polytechnique de Paris, France
[4]Soda, Inria Saclay, France
lihu.chen@imperial.ac.uk, gerard.demelo@hpi.de,
fabian.suchanek@telecom-paris.fr, gael.varoquaux@inria.fr

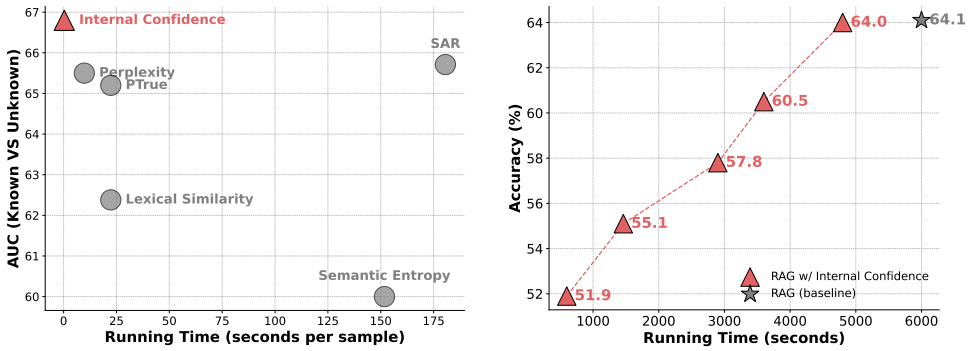
Abstract

It is important for Large Language Models (LLMs) to be aware of the boundary of their knowledge, distinguishing queries they can confidently answer from those that lie beyond their capabilities. Such awareness enables models to perform adaptive inference, such as invoking retrieval-augmented generation (RAG), engaging in slow and deep thinking, or abstaining from answering when appropriate. These mechanisms are key to developing efficient and trustworthy AI. In this work, we propose a method to detect knowledge boundaries via **Query-Level Uncertainty**, which estimates if a model is capable of answering a given query *before* generating any tokens, thus avoiding the generation cost. To this end, we propose a novel, training-free method called **Internal Confidence**, which leverages self-evaluations across layers and tokens to provide a reliable signal of uncertainty. Empirical studies on both factual question answering and mathematical reasoning tasks demonstrate that our Internal Confidence outperforms several baselines in quality of confidence while being computationally cheaper. Furthermore, we demonstrate its benefits in adaptive inference settings, showing that for RAG and model cascading it reduces inference costs while preserving overall performance.

1 Introduction

Large language Models (LLMs) have their knowledge boundaries [Li et al., 2024, Yin et al., 2024, Ren et al., 2025], which means that there are certain problems for which they cannot provide accurate answers. It is crucial for LLMs to be self-aware of their limitations, i.e., *to know what they know and know what they do not know* [Kadavath et al., 2022, Amayuelas et al., 2024].

Clear awareness of knowledge boundaries is central to improving AI, both for efficiency and trustworthiness. The rising usage of LLMs and agents has introduced significant computational and monetary costs [Varoquaux et al., 2025]. For example, agentic workflows may cost 5×–25× more per query compared to a simpler LLM prompt [Anthropic, 2025]. Regarding efficiency, if LLMs can distinguish known from unknown or simple from hard queries, they can smartly perform *adaptive inference* to navigate the trade-offs between computational cost and output quality [Chen and Varoquaux, 2024]. For queries beyond their parametric knowledge, they can actively trigger RAG to obtain external knowledge [Lewis et al., 2020] or tool calls [Schick et al., 2023]. When faced with hard problems, LLMs can engage in slow (or deep) thinking to improve their outputs, which is also known as test-time scaling [Snell et al., 2024, Zhang et al., 2025]. Alternatively, they can defer a complex problem to a larger model via model cascading [Dohan et al., 2022, Gupta et al., 2024]. This adaptive inference ensures efficient allocation of computational resources, reducing costs while maintaining performance, especially for agentic scenarios. Beyond efficiency, estimating whether a query is answerable also enhances honesty and trustworthiness of LLMs. When faced with highly uncertain



(a) Comparison of performance and running time between our query-level Internal Confidence method and existing answer-level uncertainty measures (Qwen-14B on GSM8K).

(b) Trade-off between running time and performance under different Internal Confidence thresholds for deciding on RAG invocation (Phi-3.8B on TriviaQA) compared against always using RAG.

Figure 1: Our *Internal Confidence* method improves performance / running time tradeoffs in factuality assessment and RAG settings.

queries, models can adopt an abstention strategy [Wen et al., 2024] to withhold potentially misleading responses, important in high-stakes domains like healthcare [Tomani et al., 2024].

In this work, we introduce the concept of *Query-Level Uncertainty* to estimate a model's knowledge with regard to a given query. The central research question here is: *Given a query, can we determine whether the model can address it before generating any tokens?* Most existing work focuses on answer-level uncertainty, which measures the uncertainty associated with a specific answer and is commonly used to assess the reliability of model outputs [Shorinwa et al., 2024, Vashurin et al., 2025]. In contrast, our approach shifts from post-generation to pre-generation, measuring how confidently an LLM can solve a given query, prior to answer generation, as illustrated in Figure 2. This approach avoids the computational cost of generating potentially long answers.

Prior research has explored different strategies for uncertainty estimation. One line of work learns a probe of internal states to predict uncertainties of queries [Gottesman and Geva, 2024, Kossen et al., 2024]. Another branch of work attempts to teach LLMs to explicitly express "I don't know" in their responses via fine-tuning methods [Amayuelas et al., 2024, Kapoor et al., 2024, Cohen et al., 2024, Zhang et al., 2024a]. One common issue of these studies is that they require fine-tuning and training samples, which introduces additional overhead and may restrict their generalizability across models

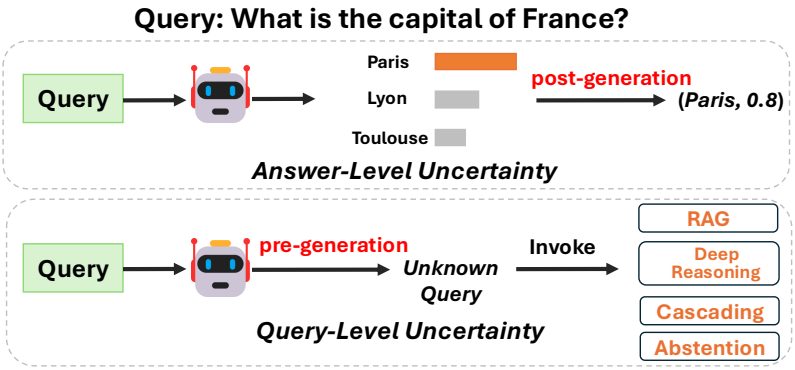


Figure 2: Illustrating the difference between answer-level and query-level uncertainty. Query-level uncertainty estimation distinguishes known from unknown queries (*knowledge boundary*) before generating answers, which is useful for adaptive inference, e.g., efficient RAG, fast–slow reasoning, or cascading models with different abilities.

and domains. To address this gap, we introduce a training-free approach to estimate query-level uncertainty that is both simple and effective.

Our approach, termed *Internal Confidence*, leverages self-evaluation across internal layers and tokens. It is grounded in a simple assumption: LLMs can internally self-assess the boundaries of their knowledge through a single forward pass over the given query, without generating an explicit answer. Inspired by the uncertainty measure P(TRUE) [Kadavath et al., 2022], we prompt LLMs with a yes–no question to self-assess if they are capable of answering a given query, and define the probability assigned to the token YES as the confidence level, denoted as P(YES). To fully exploit the latent knowledge within LLMs, our improved Internal Confidence approach computes this sort of P(YES) at each layer and token position. Subsequently, we aggregate these signals to obtain the overall confidence score. This aggregation is motivated by prior work showing that leveraging logical consistency across layers can improve outputs [Burns et al., 2022, Chuang et al., 2023, Xie et al., 2024]. Concretely, we compute a weighted sum across layers and tokens, and the weights are derived from attenuated encoding [Chen et al., 2023], which enables fine-grained control of the influence of adjacent units.

To validate the effectiveness of our proposed Internal Confidence, we conduct experiments on three datasets that cover factual QA and mathematical reasoning tasks. For fair comparison, we adapt existing answer-level methods to the query level. Experimental results demonstrate that our proposed Internal Confidence can distinguish between known and unknown queries more accurately than a range of baselines, while being substantially faster than answer-level approaches (Figure 1a). In terms of applications, we showcase that our proposed method can support efficient RAG and model cascading. On the one hand, Internal Confidence can guide users to assess the trade-offs between cost and quality when invoking additional services. On the other hand, it reveals an "optimal point", where inference overhead can be reduced without compromising performance (Figure 1b). In conclusion, we introduce the notion of query-level uncertainty and propose a simple yet effective training-free method to estimate it, which enables models to determine whether a query can be addressed without generating any tokens.

2 Problem Statement and Method

In this section, we define the problem and introduce our method, *Internal Confidence*, a score that reflects whether an LLM can address a query in its own knowledge, prior to generating tokens.

2.1 Problem Statement

Given a query (including prompt tokens) $\mathbf{x} = (x_1, \ldots, x_N)$, we aim to quantify the query-level uncertainty, $U(\mathbf{x})$, without generating an answer $\mathbf{y}$. This differs from existing uncertainty approaches that estimate the uncertainty associated with a specific generated answer, an answer-level uncertainty that can be denoted as $U(\mathbf{x}, \mathbf{y})$. We define a query as being within the model's knowledge boundary if the LLM can produce a correct answer under greedy decoding, i.e., by selecting the highest-probability token at each step without sampling. Conversely, failure to produce the correct answer suggests the query falls beyond the model's boundary, and it does not possess sufficient knowledge to answer it. While greedy decoding ensures deterministic measurement, it may not always reflect the optimal performance of a model [Song et al., 2024], as alternative decoding strategies like beam search may elicit a better answer. Therefore, this pragmatic framework serves as a heuristic indicator of internal knowledge, rather than an absolute measure. We use this standard to evaluate the estimated query-level uncertainty, i.e., a lower uncertainty indicates a model is more likely to output the correct answer.

Our problem formulation mostly targets epistemic uncertainty of the model, though specific queries and datasets may contain aleatoric effects (see details in Section A). Our study focuses on queries with definite and clear-cut answers, as in factual QA and mathematical reasoning, which have broad applications and allow for clear evaluations. While contentious queries with open and subjective answers are also important in areas such as politics and philosophy, they remain beyond the scope of this work.

2.2 Method: From P(YES) to Internal Confidence

Studies have revealed that LLMs can express verbalized uncertainty in their responses [Tian et al., 2023, Xiong et al., 2024], which indicates that LLMs possess an internal mechanism for assessing the correctness of their outputs. Building on this observation, one can explicitly prompt an LLM to self-assess its confidence in answering a given query by constraining the response to a yes–no binary format: *"Respond only with 'Yes' or 'No' to indicate whether you are capable of answering the* `{Query}` *accurately. Answer Yes or No:"*. Following that, we can compute the probability assigned to the token P(YES) at the last token (x_N):

$$\mathrm{P}(\textsc{Yes}) = \mathrm{softmax}\big(\mathbf{W}^{\text{unemb}}_{[\textsc{Yes},\textsc{No}]}\,\mathbf{h}_N^{(L)}\big)_{\textsc{Yes}} \tag{1}$$

Here, N is the index of the last token in the query and L is the index of the last layer of the model. $\mathbf{h}_N^{(L)} \in \mathbb{R}^d$ is the hidden state, where d is the dimensionality of the hidden representations. $\mathbf{W}^{\text{unemb}} \in \mathbb{R}^{|\mathcal{V}|\times d}$ is the unembedding matrix that maps the hidden state $\mathbf{h}_N^{(L)}$ to logits over the vocabulary $\mathcal{V}$. The probability P(YES) can serve as a query-level confidence score here, which is similar to the process of linear probing [Alain and Bengio, 2016], but without any training steps. While this measure is correlated with verbalized uncertainty, a key distinction is that it requires only a single forward pass of the query, without generating any answer tokens.

However, P(YES) considers only the final hidden state of the LLM, although the intermediate internal states of LLMs preserve rich knowledge and latent information [Chen et al., 2025], especially for uncertainty estimation [Azaria and Mitchell, 2023, Chen et al., 2024a]. Furthermore, prior work demonstrates that incorporating logical consistency across layers can improve outputs [Burns et al., 2022, Chuang et al., 2023, Xie et al., 2024].

Motivated by these insights, we propose the *Internal Confidence*, a method that leverages latent knowledge distributed across multiple layers and tokens. Formally, let f_θ denote the transformation function for computing hidden states, parametrized by θ. The hidden state for the token x_n of the input query at layer l is computed as:

$$\mathbf{h}_n^{(l)} = f_\theta(\mathbf{h}_1^{(l-1)}, \ldots, \mathbf{h}_n^{(l-1)}) \tag{2}$$

In total, the model contains $N \times L$ such latent representations, and we can use Equation 1 to compute the P(YES) for each $\mathbf{h}_n^{(l)}$.

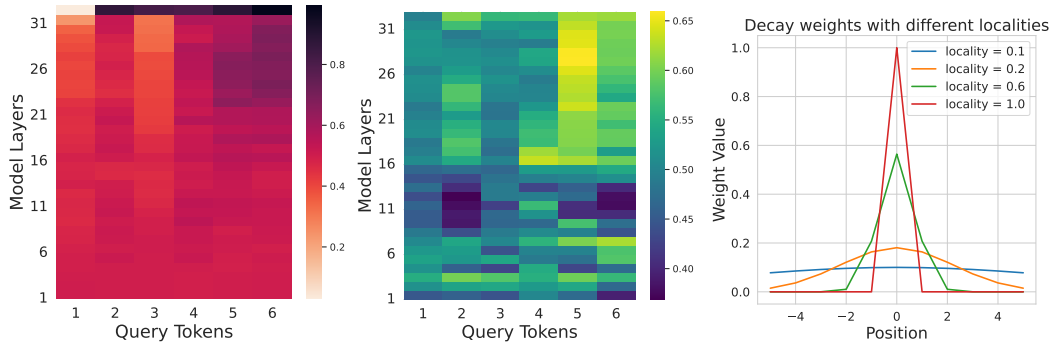
Figure 3a plots the average P(YES) of Llama-8B on mathematical queries (the validation set of GSM8K [Cobbe et al., 2021]), across layers and query tokens.[1] We observe that the P(YES) generally increases from lower to higher layers and from left to right positions. If we treat each $\mathrm{P}(\textsc{Yes} \mid \mathbf{h}_n^{(l)})$ as a confidence score and compute the Area Under the Curve (AUC), we can obtain an AUC heatmap that illustrates how effectively each internal representation can distinguish known and unknown queries. As shown in Figure 3b, the highest score does not necessarily appear at the top right position. Instead, the representation $\mathbf{h}_5^{(27)}$ yields the best AUC, and the performance gradually declines in regions surrounding this point. We refer to this optimal point as the *decision center*, where the model most effectively separates known from unknown queries.

To improve the vanilla P(YES), we can apply weighted average centering around the decision center, which serves as an ensemble strategy to enhance calibration and expressivity [Zhang et al., 2020, Stickland and Murray, 2020]. We refer to this process as *Internal Confidence (IC)*, formally defined as:

$$\mathrm{IC}(\mathbf{h}) = \sum_{n=1}^{N}\sum_{l=1}^{L} w_n^{(l)}\,\mathrm{P}(\textsc{Yes} \mid \mathbf{h}_n^{(l)}), \tag{3}$$

where $w_n^{(l)}$ denotes the weight assigned to the hidden representation $\mathbf{h}_n^{(l)}$. The equation describes a hierarchical two-step aggregation process. In the first step, for each individual token, we compute a weighted sum of confidence scores across layers. In the second step, we aggregate these token-level scores using another weighted average. Conceptually, this process can be parameterized by a layer weight vector $\mathbf{w}^{\text{layer}} \in \mathbb{R}^L$ for the first step and a token weight vector $\mathbf{w}^{\text{token}} \in \mathbb{R}^N$ for the second step.

[1] Here, we consider the last k tokens of a query, assuming that a model has seen the entire query and is able to infer its knowledge gap.



(a) P(YES) (b) AUC (c) Decay Weights

Figure 3: **Left:** the internal P(YES) across tokens and layers. **Middle:** the AUC of P(YES) across tokens and layers. **Right:** decay weights with different localities. Model: Llama-8B; Dataset: GSM8K validation set.

The obtained IC($\mathbf{h}$) value provides a single, refined confidence score that integrates rich information across both layers and tokens.

In our implementation, we adopt the top-right cell (corresponding to the last token and last layer) as the decision center, since we observe that the decision center tends to be located near the later layers and final tokens across various architectures and tasks. While, in principle, the optimal decision center may also lie elsewhere, identifying such an optimal center would require a hold-out set of training data, which conflicts with our goal of developing a training-free approach. To address this, rather than relying on model- or task-specific tuning of the decision center, we incorporate information from the neighborhood of the fixed top-right cell. This strategy allows us to have the potential benefits of the optimal decision center while maintaining generalizability and avoiding dependence on additional training samples.

To reflect the observation that the AUC performance gradually decays away from the decision center, we adopt Attenuated Encoding, as proposed by Chen et al. [2023], to compute the above weight vectors in Equation 3:

$$\delta_{i,j} = \frac{\exp(-\alpha\ |i-j|^2)}{\sum_{j=1}^{J} \exp(-\alpha\ |i-j|^2)}, \tag{4}$$

where i is the index of the decision center, $|i-j|$ is the relative distance, and $\alpha > 0$ is a scalar parameter that controls the locality value. Locality is a metric that measures the extent to which weights are concentrated in adjacent positions of a center. Given a weight vector $\epsilon = \{\epsilon_1, \epsilon_2, ..., \epsilon_J\}$ and assuming that the center index is i, the locality can be expressed as:

$$\text{Loc}(\epsilon) \in [0,1] = \sum_{j=1}^{J} \frac{\epsilon_j}{2^{|i-j|}} \tag{5}$$

Here, a value of 1 implies that the vector perfectly satisfies the locality property. Figure 3c plots the weights obtained from Equation 4 for varying degrees of locality. This shows that we can account for the influence of neighboring layers and tokens during the averaging process.

Our proposed Internal Confidence is training-free and computationally efficient, as it requires only a single forward pass for a given query. Since model responses are frequently longer than input prompts and invoking external services such as RAG and deep thinking adds significant overhead, we propose this pre-generation uncertainty to support adaptive reasoning.

3 Experiments

3.1 Settings

Models. Our experiments consider three different LLM sizes: *Phi-3-mini-4k-instruct* [Abdin et al., 2024], *Llama-3.1-8B-Instruct* [Grattafiori et al., 2024], and *Qwen2.5-14B-Instruct* [Team, 2024].

Method	TriviaQA			SciQ			GSM8K			Avg		
	↑ AUC	↑ PRR	↓ ECE	↑ AUC	↑ PRR	↓ ECE	↑ AUC	↑ PRR	↓ ECE	↑ AUC	↑ PRR	↓ ECE
Phi-3.8B												
Max$(-\log p)$	55.5	10.0	—	51.4	2.9	—	55.0	11.3	—	54.0	8.1	—
Predictive Entropy	58.9	17.9	—	51.2	3.9	—	**63.6**	**25.7**	—	57.9	15.8	—
Min-K Entropy	59.9	20.0	—	52.7	4.9	—	60.4	17.9	—	57.7	14.3	—
Attentional Entropy	60.6	21.4	—	56.2	9.4	—	52.4	4.4	—	56.4	11.7	—
Perplexity	61.8	24.3	—	57.7	16.6	—	53.6	6.9	—	57.7	15.9	—
Internal Semantic Similarity	48.7	-2.4	0.3	46.9	-5.9	12.2	47.9	-2.6	35.2	47.8	-3.6	15.9
P(YES) (*top right*)	**64.9**	27.7	**5.4**	**61.3**	24.4	**5.9**	53.3	9.4	**11.3**	**59.8**	20.5	**7.5**
P(YES) (*naive avg*)	64.1	28.3	17.0	57.5	18.8	6.4	50.5	9.3	25.4	57.4	18.8	16.3
Internal Confidence	64.7	**30.1**	7.9	60.7	**25.8**	10.4	53.9	6.4	19.9	**59.8**	**20.8**	12.7
Llama-8B												
Max$(-\log p)$	54.9	11.1	—	51.4	1.9	—	53.3	10.4	—	53.2	7.8	—
Predictive Entropy	58.5	17.7	—	51.4	3.2	—	**66.1**	28.0	—	58.7	16.3	—
Min-K Entropy	58.1	17.4	—	53.5	7.9	—	57.5	13.2	—	56.4	12.8	—
Attentional Entropy	59.4	18.7	—	57.7	15.2	—	56.1	13.5	—	57.7	15.8	—
Perplexity	58.6	17.1	—	58.3	15.1	—	53.2	4.3	—	56.7	12.2	—
Internal Semantic Similarity	44.1	-14.4	24.4	46.1	-7.1	30.8	52.7	6.7	45.9	47.6	-4.9	33.7
P(YES) (*top right*)	55.4	10.2	31.7	**58.4**	**17.2**	23.7	52.6	5.2	11.9	55.5	10.9	22.4
P(YES) (*naive avg*)	65.9	33.0	**12.6**	57.9	14.9	20.4	61.3	18.5	33.5	61.7	22.1	22.2
Internal Confidence	**68.7**	**35.5**	25.4	58.1	15.7	**16.7**	65.7	**34.9**	**3.1**	**64.2**	**28.7**	**15.1**
Qwen-14B												
Max$(-\log p)$	56.5	12.4	—	54.1	6.9	—	54.3	13.5	—	55.0	10.9	—
Predictive Entropy	59.3	18.9	—	53.2	6.9	—	66.4	**32.6**	—	59.6	19.5	—
Min-K Entropy	59.9	20.0	—	55.7	11.3	—	63.0	30.9	—	59.5	20.7	—
Attentional Entropy	59.1	17.2	—	59.4	19.2	—	54.9	3.1	—	57.8	13.2	—
Perplexity	59.1	17.8	—	60.1	20.7	—	54.0	7.3	—	57.7	15.3	—
Internal Semantic Similarity	51.0	2.5	**2.0**	45.5	-7.7	14.9	47.5	-4.6	33.1	48.0	-3.3	16.7
P(YES) (*top right*)	67.8	36.0	30.3	60.0	21.7	24.1	55.0	11.7	6.4	60.9	23.1	20.3
P(YES) (*naive avg*)	67.0	33.9	3.5	59.5	17.9	**14.6**	64.0	32.3	32.4	63.5	28.0	**16.8**
Internal Confidence	**71.9**	**43.3**	26.5	**62.6**	**23.6**	18.2	**66.8**	28.2	**5.7**	**67.1**	**31.7**	**16.8**

Table 1: Overall results of different query-level uncertainty estimation methods. The best-performing methods are highlighted using boldface and second-best results are underlined.

This allows us to assess whether Internal Confidence generalizes across different model sizes. It is worth noting that Internal Confidence can also be applied to models without instruction tuning.

Implementations. For Llama and Qwen, Internal Confidence is computed in the zero-shot setting, whereas for Phi, we use three shots in the prompt, since smaller models benefit from demonstration-based guidance (See details in Section D.2). All LLMs employ greedy decoding to ensure deterministic outputs. The decision center is fixed to the last layer and last token, and we set $\alpha = 1.0$ (Equation 4) across all models and datasets.

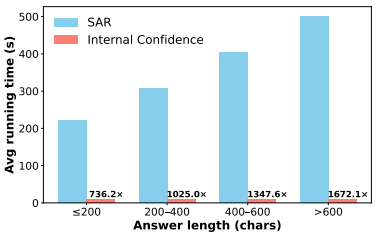


Figure 4: Acceleration ratio comparison between answer-level SAR and our Internal Confidence.

Evaluation Datasets. We evaluate on two factual QA datasets and one mathematical reasoning dataset: TriviaQA [Joshi et al., 2017], SciQ [Welbl et al., 2017], and GSM8K [Cobbe et al., 2021]. The first two tasks aim to assess factual knowledge stored in parameters, while GSM8K requires models to self-evaluate their reasoning capabilities. The ground truth for factual QA tasks takes the form of a short answer with entity-related facts. GSM8k as well calls for a short answer, but the intermediate reasoning steps are evaluated as well, following prior work [Kadavath et al., 2022]. The three datasets consist of 10,000, 10,000, and 5,000 samples, respectively, with 1,000 samples from each reserved for validation.

We elicit responses from the model using a greedy decoding strategy. If the answer aligns with the ground truth, we consider the model as possessing sufficient knowledge and the query as falling within its knowledge boundary. For the first two datasets with short answers, answers are deemed correct if the ROUGE-L [Lin and Och, 2004] of the ground truth is greater than 0.3, which is

consistent with prior work [Kuhn et al., 2023]. For the GSM8K dataset, we use an LLM evaluator, Mistral-Large [MistralAI, 2024], to assess both reasoning steps and the final answer. Subsequently, each query is paired with a binary label reflecting whether the model is capable of addressing it.

Baselines. For comparison, we adapt state-of-the-art answer-level methods to quantify the pre-generation uncertainty (see details in Section C): (1) Max$(-\log p)$ [Manakul et al., 2023], (2) Predictive Entropy [Malinin and Gales, 2021], (3) Min-K Entropy [Shi et al., 2024], (4) Attentional Entropy [Duan et al., 2024], (5) Perplexity, (6) Internal Semantic Similarity [Fomicheva et al., 2020], (7) P(YES) (*top right*), corresponding to Equation 1. (8) P(YES) (*naive avg*) is a variant of our Internal Confidence that adopts naive averaging to aggregate scores across different tokens and layers.

Evaluation Metrics. We evaluate uncertainty by assessing whether a method can distinguish *known* and *unknown* queries, which can be treated as ranking problems, i.e., a lower uncertainty means a model is more likely to know the answer to the query. Following prior work [Manakul et al., 2023, Kuhn et al., 2023], we adopt the Area Under the Curve (AUC) and Prediction Rejection Ratio (PRR) [Malinin et al., 2017] as metrics to measure this. Additionally, we compute the Expected Calibration Error (ECE) to assess the calibration of different methods.

3.2 Internal Confidence Can Identify Known and Unknown Queries

Table 1 summarizes the overall results comparing different query-level uncertainty methods. First, we can observe that our proposed Internal Confidence consistently outperforms other baselines in distinguishing known from unknown queries, as reflected in both average AUC and PRR. The advantage becomes more pronounced for larger models such as Llama-8B and Qwen-14B. For instance, on Qwen-14B, it obtains an average AUC of 67.1 and PRR of 31.7, clearly surpassing all other methods. Regarding the calibration (ECE), Internal Confidence is found to consistently achieve a lower error across models and tasks. These findings indicate the effectiveness of Internal Confidence. Finally, we note that the variants, P(YES) (*top right*) and P(YES) (*naive avg*), generally underperform the full method, which highlights the importance of the attenuated encoding and its decay weights in effectively aggregating signals from different layers and tokens.

3.3 Internal Confidence is Much Faster than Answer-Level Approaches

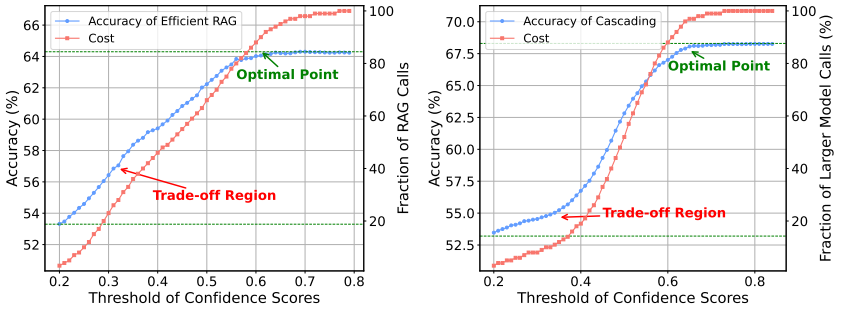
We compare our query-level Internal Confidence with several popular answer-level uncertainty methods on GSM8K using Qwen-14B, including Perplexity [Fomicheva et al., 2020], Semantic Entropy [Kuhn et al., 2023], P(TRUE) [Kadavath et al., 2022], Lexical Similarity [Fomicheva et al., 2020], and SAR [Duan et al., 2024].

Table 2 compares the effectiveness and runtime across different approaches. While answer-level approaches such as Perplexity, P(TRUE), and SAR require significantly higher computation time (ranging from nearly 10 seconds up to more than 180 seconds per sample), our Internal Confidence method achieves the best AUC (66.8) with an average running time of only 0.3 seconds. This corresponds to speedups of over 30× to 600× compared to existing baselines. These results demonstrate that Internal Confidence combines state-of-the-art accuracy with an extremely fast inference speed, which can be a practical choice for large-scale or latency-sensitive reasoning tasks.

Method	↑ AUC	↓ Time (s)	↑ Speedup
Perplexity	65.5	9.8	32×
Semantic Entropy	60.0	151.8	506×
P(TRUE)	65.2	22.3	74×
Lexical Similarity	62.4	22.3	74×
SAR	65.7	180.6	602×
Internal Confidence	**66.8**	**0.3**	—

Table 2: Comparison of query-level Internal Confidence with answer-level uncertainty methods (Qwen-14B on GSM8K).

Notably, the running time for Internal Confidence remains constant, independent of the length of answers. Figure 4 shows that the runtime of the best answer-level approach, SAR, grows with the answer length, reaching nearly 500s for answers over 600 characters. In contrast, Internal Confidence achieves large acceleration ratios (736×–1672×), with speedups increasing as answers become longer, which demonstrates its scalability and efficiency. See results of other datasets in Table A1.



(a) Efficient RAG (see the running time in Figure 1b)

(b) Model Cascading

Figure 5: **Left:** We use estimated Internal Confidence to decide whether to invoke RAG. If the Internal Confidence exceeds a threshold, the model answers the query using its parametric knowledge. Otherwise, it relies on external knowledge. The plot shows the accuracy of Phi-3.8B on the TriviaQA dataset under this setting. **Right:** We implement a model cascading setting with Phi-3.8B (small) and Llama-8B (large) on the TriviaQA dataset. The Internal Confidence of the smaller model determines whether it answers the query or defers to the larger model when confidence is low. The green lines indicate the baseline accuracy achieved by the simple model or complex model.

3.4 Internal Confidence Makes LLM Reasoning More Efficient

Recent studies advance LLM reasoning by introducing additional resources, such as using RAG to obtain external knowledge [Lewis et al., 2020] and inference-time scaling to improve outputs [Snell et al., 2024]. However, it is not always necessary to use additional resources, especially for simple queries. Here, we use our proposed Internal Confidence for adaptive inference, determining when to invoke RAG, slow thinking, or model cascading.

We conduct experiments for two scenarios: (1) *Efficient RAG.* Basically, the Internal Confidence can serve as a signal of the knowledge gaps of a model. If the score is greater than a threshold, the model is confident to address the query. Otherwise, it requires the call of RAG. We use the TriviaQA dataset for evaluation. This dataset provides web search results for a query, which can be used as retrieved contexts for RAG. (2) *Model Cascading.* This task aims to achieve cost-performance trade-offs by coordinating small and large models [Dohan et al., 2022, Gupta et al., 2024]. The smaller models are responsible for easy assignments. If they are aware that the mission is hard to complete, they invoke a larger model. We use a two-model cascade setting with Phi-3.8B and Llama-8B on the TriviaQA dataset. If the Internal Confidence of the smaller model is high, we do not invoke the larger model. Otherwise, the hard query is deferred to the larger model.

Figure 5 presents the results of applying Internal Confidence scores to efficient RAG (left) and model cascading (right). In both cases, the *trade-off region* illustrates how adjusting the confidence threshold allows us to balance efficiency and performance by controlling the frequency of external service calls or larger model invocations. The *optimal point* highlights thresholds where additional resource usage can be reduced without sacrificing accuracy. Results across the two tasks further confirm the effectiveness of Internal Confidence in identifying knowledge gaps. Our method offers practical benefits by reducing inference overhead, which can be applied to token-heavy agentic frameworks.

4 Conclusion

In this work, we propose the new notion of query-level uncertainty, which seeks to assess whether a model can successfully address a query without generating any tokens. To this end, we propose the novel Internal Confidence technique, which leverages latent self-evaluation to identify the boundary of a model's knowledge. Extensive experimental results confirm the effectiveness of our approach on both factual QA and mathematical reasoning. Our method is capable of identifying knowledge gaps with a substantially faster speed compared to answer-level approaches. Furthermore, we apply Internal Confidence to two practical scenarios of adaptive inference, efficient RAG and model cascading. Our

findings reveal that our method can identify two regions: a trade-off region and an optimal point. The former means that one can strike a balance between cost and quality by carefully selecting a threshold of confidence scores. The latter means that one can reduce inference overhead without compromising performance.

In conclusion, these results highlight Internal Confidence as a strong and general-purpose baseline for estimating query-level uncertainty. While there remains room for refinement, our study can serve as a strong baseline for this task, and we hope this study can stimulate future studies in this area.

References

M. Abdin, J. Aneja, H. Awadalla, A. Awadallah, A. A. Awan, N. Bach, A. Bahree, A. Bakhtiari, J. Bao, H. Behl, et al. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*, 2024.

G. Alain and Y. Bengio. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016.

A. Amayuelas, K. Wong, L. Pan, W. Chen, and W. Y. Wang. Knowledge of knowledge: Exploring known-unknowns uncertainty with large language models. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 6416–6432, 2024.

Anthropic. The cost of thinking: Agentic ai, inference economics, and the future of hybrid intelligence. `https://medium.com/@ahilanp/the-cost-of-thinking-agentic-ai-inference-economics-and-the-future-of-hybrid-intelligence-139` 2025.

A. Azaria and T. Mitchell. The internal state of an llm knows when it's lying. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 967–976, 2023.

C. Burns, H. Ye, D. Klein, and J. Steinhardt. Discovering latent knowledge in language models without supervision. In *The Eleventh International Conference on Learning Representations*, 2022.

C. Chen, K. Liu, Z. Chen, Y. Gu, Y. Wu, M. Tao, Z. Fu, and J. Ye. Inside: Llms' internal states retain the power of hallucination detection. In *ICLR*, 2024a.

L. Chen and G. Varoquaux. What is the role of small models in the llm era: A survey. *arXiv preprint arXiv:2409.06857*, 2024.

L. Chen, G. Varoquaux, and F. Suchanek. The locality and symmetry of positional encodings. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 14313–14331, 2023.

L. Chen, A. Perez-Lebel, F. Suchanek, and G. Varoquaux. Reconfidencing llms from the grouping loss perspective. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 1567–1581, 2024b.

L. Chen, A. Dejl, and F. Toni. Identifying query-relevant neurons in large language models for long-form texts. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 23595–23604, 2025.

Y.-S. Chuang, Y. Xie, H. Luo, Y. Kim, J. R. Glass, and P. He. Dola: Decoding by contrasting layers improves factuality in large language models. In *The Twelfth International Conference on Learning Representations*, 2023.

K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

R. Cohen, K. Dobler, E. Biran, and G. de Melo. I don't know: Explicit modeling of uncertainty with an [idk] token. *Advances in Neural Information Processing Systems*, 37:10935–10958, 2024.

A. Der Kiureghian and O. Ditlevsen. Aleatory or epistemic? does it matter? *Structural safety*, 31(2): 105–112, 2009.

D. Dohan, W. Xu, A. Lewkowycz, J. Austin, D. Bieber, R. G. Lopes, Y. Wu, H. Michalewski, R. A. Saurous, J. Sohl-Dickstein, et al. Language model cascades. *arXiv preprint arXiv:2207.10342*, 2022.

J. Duan, H. Cheng, S. Wang, A. Zavalny, C. Wang, R. Xu, B. Kailkhura, and K. Xu. Shifting attention to relevance: Towards the predictive uncertainty quantification of free-form large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5050–5063, 2024.

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*, 8:539–555, 2020.

J. Geng, F. Cai, Y. Wang, H. Koeppl, P. Nakov, and I. Gurevych. A survey of confidence estimation and calibration in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6577–6595, 2024.

B. S. Gillon. Ambiguity, generality, and indeterminacy: Tests and definitions. *Synthese*, 85(3): 391–416, 1990.

D. Gottesman and M. Geva. Estimating knowledge in large language models without generating a single token. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 3994–4019, 2024.

A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

N. Gupta, H. Narasimhan, W. Jitkrittum, A. S. Rawat, A. K. Menon, and S. Kumar. Language model cascades: Token-level uncertainty and beyond. In *The Twelfth International Conference on Learning Representations*, 2024.

S. C. Hora. Aleatory and epistemic uncertainty in probability elicitation with an example from hazardous waste management. *Reliability Engineering & System Safety*, 54(2-3):217–223, 1996.

E. Hüllermeier and W. Waegeman. Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. *Machine learning*, 110(3):457–506, 2021.

M. Joshi, E. Choi, D. S. Weld, and L. Zettlemoyer. 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, 2017.

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.

S. Kapoor, N. Gruver, M. Roberts, K. M. Collins, A. Pal, U. Bhatt, A. Weller, S. Dooley, M. Goldblum, and A. G. Wilson. Large language models must be taught to know what they don't know. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

J. Kossen, J. Han, M. Razzak, L. Schut, S. Malik, and Y. Gal. Semantic entropy probes: Robust and cheap hallucination detection in llms. *arXiv preprint arXiv:2406.15927*, 2024.

L. Kuhn, Y. Gal, and S. Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *Proceedings of the Eleventh International Conference on Learning Representations (ICLR)*, Kigali, Rwanda, May 1–5 2023. OpenReview.net. URL `https://openreview.net/forum?id=O3d9M3hNya`.

P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474, 2020.

M. Li, Y. Zhao, Y. Deng, W. Zhang, S. Li, W. Xie, S.-K. Ng, and T.-S. Chua. Knowledge boundary of large language models: A survey. *arXiv preprint arXiv:2412.12472*, 2024.

C.-Y. Lin and F. J. Och. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *Proceedings of the 42nd annual meeting of the association for computational linguistics (ACL-04)*, pages 605–612, 2004.

Z. Lin, S. Trivedi, and J. Sun. Generating with confidence: Uncertainty quantification for black-box large language models. *Transactions on Machine Learning Research*, 2023.

M. Mahaut, L. Aina, P. Czarnowska, M. Hardalov, T. Mueller, and L. Màrquez. Factual confidence of llms: on reliability and robustness of current estimators. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4554–4570, 2024.

A. Malinin and M. Gales. Uncertainty estimation in autoregressive structured prediction. In *International Conference on Learning Representations*, 2021.

A. Malinin, A. Ragni, K. Knill, and M. Gales. 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, 2017.

P. Manakul, A. Liusie, and M. Gales. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9004–9017, 2023.

MistralAI. Mistral large: A general-purpose language model. `https://mistral.ai/news/mistral-large-2407/`, 2024.

B. Mucsányi, M. Kirchhof, and S. J. Oh. Benchmarking uncertainty disentanglement: Specialized uncertainties for specialized tasks. *Advances in neural information processing systems*, 37:50972–51038, 2024.

R. Ren, Y. Wang, Y. Qu, W. X. Zhao, J. Liu, H. Wu, J.-R. Wen, and H. Wang. Investigating the factual knowledge boundary of large language models with retrieval augmentation. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 3697–3715, 2025.

T. Schick, J. Dwivedi-Yu, R. Dessì, R. Raileanu, M. Lomeli, E. Hambro, L. Zettlemoyer, N. Cancedda, and T. Scialom. Toolformer: Language models can teach themselves to use tools. *Advances in Neural Information Processing Systems*, 36:68539–68551, 2023.

W. Shi, A. Ajith, M. Xia, Y. Huang, D. Liu, T. Blevins, D. Chen, and L. Zettlemoyer. Detecting pretraining data from large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

O. Shorinwa, Z. Mei, J. Lidard, A. Z. Ren, and A. Majumdar. A survey on uncertainty quantification of large language models: Taxonomy, open research challenges, and future directions. *arXiv preprint arXiv:2412.05563*, 2024.

C. Snell, J. Lee, K. Xu, and A. Kumar. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024.

Y. Song, G. Wang, S. Li, and B. Y. Lin. The good, the bad, and the greedy: Evaluation of llms should not ignore non-determinism. *arXiv preprint arXiv:2407.10457*, 2024.

A. C. Stickland and I. Murray. Diverse ensembles improve calibration. *arXiv preprint arXiv:2007.04206*, 2020.

Q. Team. Qwen2.5: A party of foundation models, September 2024. URL `https://qwenlm.github.io/blog/qwen2.5/`.

K. Tian, E. Mitchell, A. Zhou, A. Sharma, R. Rafailov, H. Yao, C. Finn, and C. D. Manning. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5433–5442, 2023.

C. Tomani, K. Chaudhuri, I. Evtimov, D. Cremers, and M. Ibrahim. Uncertainty-based abstention in llms improves safety and reduces hallucinations. *arXiv preprint arXiv:2404.10960*, 2024.

G. Varoquaux, S. Luccioni, and M. Whittaker. Hype, sustainability, and the price of the bigger-is-better paradigm in ai. In *Proceedings of the 2025 ACM Conference on Fairness, Accountability, and Transparency*, pages 61–75, 2025.

R. Vashurin, E. Fadeeva, A. Vazhentsev, L. Rvanova, D. Vasilev, A. Tsvigun, S. Petrakov, R. Xing, A. Sadallah, K. Grishchenkov, et al. Benchmarking uncertainty quantification methods for large language models with lm-polygraph. *Transactions of the Association for Computational Linguistics*, 13:220–248, 2025.

H. Wang, B. Xue, B. Zhou, T. Zhang, C. Wang, H. Wang, G. Chen, and K.-f. Wong. Self-dc: When to reason and when to act? self divide-and-conquer for compositional unknown questions. *arXiv preprint arXiv:2402.13514*, 2024.

J. Welbl, N. F. Liu, and M. Gardner. Crowdsourcing multiple choice science questions. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 94–106, 2017.

B. Wen, J. Yao, S. Feng, C. Xu, Y. Tsvetkov, B. Howe, and L. L. Wang. Know your limits: A survey of abstention in large language models. *arXiv preprint arXiv:2407.18418*, 2024.

Z. Xie, J. Guo, T. Yu, and S. Li. Calibrating reasoning in language models with internal consistency. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

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. In *The Twelfth International Conference on Learning Representations*, 2024.

X. Yin, X. Zhang, J. Ruan, and X. Wan. Benchmarking knowledge boundary for large language models: A different perspective on model evaluation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2270–2286, 2024.

H. Zhang, S. Diao, Y. Lin, Y. Fung, Q. Lian, X. Wang, Y. Chen, H. Ji, and T. Zhang. R-tuning: Instructing large language models to say 'i don't know'. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7106–7132, 2024a.

J. Zhang, B. Kailkhura, and T. Y.-J. Han. Mix-n-match: Ensemble and compositional methods for uncertainty calibration in deep learning. In *International conference on machine learning*, pages 11117–11128. PMLR, 2020.

M. Zhang, M. Huang, R. Shi, L. Guo, C. Peng, P. Yan, Y. Zhou, and X. Qiu. Calibrating the confidence of large language models by eliciting fidelity. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 2959–2979, 2024b.

Q. Zhang, F. Lyu, Z. Sun, L. Wang, W. Zhang, W. Hua, H. Wu, Z. Guo, Y. Wang, N. Muennighoff, et al. A survey on test-time scaling in large language models: What, how, where, and how well? *arXiv preprint arXiv:2503.24235*, 2025.

A Fundamental Concepts

A.1 Aleatoric and Epistemic Uncertainty

Uncertainty in machine learning is commonly categorized into two main types: aleatoric and epistemic uncertainty [Hora, 1996, Der Kiureghian and Ditlevsen, 2009, Hüllermeier and Waegeman, 2021]. These distinctions are often overlooked in the context of LLM uncertainty estimation. Aleatoric uncertainty arises from inherent randomness in the data, such as ambiguous inputs or conflicting

annotations. This type of uncertainty is irreducible, as it reflects intrinsic noise in the input data. In contrast, epistemic uncertainty stems from a lack of knowledge, often due to insufficient training data and limited model capacity. Unlike aleatoric uncertainty, epistemic uncertainty is reducible with additional data or advanced modeling. In this work, we focus specifically on epistemic uncertainty, with the goal of evaluating whether an LLM possesses sufficient knowledge to answer a given query. For evaluation, we adopt factual QA and mathematical reasoning benchmarks, which are designed to have clear-cut answers. We assume these datasets are well-curated to minimize aleatoric uncertainty, such as ambiguous questions and inconsistent labels. However, we acknowledge that residual ambiguity may persist, given the inherent nature of linguistic ambiguity [Gillon, 1990] and the difficulty of fully disentangling aleatoric from epistemic uncertainty [Mucsányi et al., 2024]. We treat such aleatoric effects as negligible for the purposes of focusing on epistemic uncertainty.

A.2 Uncertainty and Confidence

In the context of LLMs, the terms uncertainty and confidence are often used interchangeably (as antonyms). However, the two concepts have subtle differences. As noted by Lin et al. [2023], uncertainty is a holistic property of the entire predictive distribution, while confidence refers to the model's estimated confidence level associated with a specific answer. For example, given a query $x =$*"What is the capital of France"*, estimating uncertainty conceptually requires the distribution over all plausible answers, e.g., *Paris*, *Toulouse*, *Lyon*, etc., as operationalized by the semantic entropy framework [Kuhn et al., 2023], which clusters semantically equivalent outputs before computing entropy. In contrast, the conditional probability $\mathrm{P}(Y = \text{Paris} \mid x)$ can serve as an indication of confidence here, reflecting how strongly the model supports that particular response. Given that it is unfeasible to enumerate all possible responses in our context of query-level uncertainty, we pragmatically treat uncertainty and confidence as antonyms.

B Related Work

B.1 Uncertainty Estimation and LLMs

Existing approaches to LLM uncertainty primarily focus on estimating the uncertainty of LLM-generated responses, by providing a score intended to reflect the reliability of a query–answer pair [Geng et al., 2024, Shorinwa et al., 2024, Mahaut et al., 2024, Vashurin et al., 2025]. These approaches often rely on internal states [Chen et al., 2024a] or textual responses [Kuhn et al., 2023], and commonly use calibration techniques to mitigate issues such as overconfidence [Zhang et al., 2024b] and biases [Chen et al., 2024b]. Notably, these methods assess *post-generation* reliability, i.e., uncertainty regarding a specific answer after it has been produced. In contrast, relatively little work has explored how to quantify a model's ability to address a query prior to token generation. For example, Gottesman and Geva [2024] propose training a lightweight probe on internal representations to estimate the model's knowledge about specific entities. Similarly, Semantic Entropy Probes [Kossen et al., 2024] suggest that internal model states can implicitly encode semantic uncertainty, even before any output is generated. To the best of our knowledge, this work is the first to formally define query-level uncertainty and to investigate it systematically.

B.2 Knowledge Boundary Detection

LLMs should be able to faithfully assess their level of confidence in answering a query. This awareness of knowledge boundaries [Li et al., 2024, Yin et al., 2024, Wang et al., 2024] is essential for building reliable AI systems, particularly in high-stakes domains such as healthcare and law. A pioneering study by Kadavath et al. [2022] explores whether language models can be trained to predict when they "know" the answer to a given query, introducing the concept of "I Know" (IK) prediction. Based on this idea, subsequent work has proposed methods to help LLMs become explicitly aware of their knowledge limitations through fine-tuning strategies [Amayuelas et al., 2024, Kapoor et al., 2024]. Cohen et al. [2024] further advances this line of research by introducing a special `[IDK]` ("*I don't know*") token into the model's vocabulary, allowing the direct expression of uncertainty in its output. Similarly, R-Tuning [Zhang et al., 2024a] tunes LLMs to refrain from responding to questions beyond their parametric knowledge. While these abstention-based approaches show benefits in mitigating hallucinations [Wen et al., 2024], they often require additional fine-tuning, which introduces overhead and may limit generalizability across models and tasks. In this work, we

Method	↑ AUC	↓ Time (s)	↑ Speedup
TriviaQA			
Perplexity	75.1	5.6	28×
Semantic Entropy	72.3	139.5	698×
P(TRUE)	65.2	22.5	113×
Lexical Similarity	**77.2**	142.3	712×
SAR	76.5	160.8	804×
Internal Confidence	71.9	**0.2**	—
SciQ			
Perplexity	**71.5**	12.9	65×
Semantic Entropy	66.3	132.8	664×
P(TRUE)	60.4	22.1	111×
Lexical Similarity	68.7	165.1	826×
SAR	70.5	165.7	829×
Internal Confidence	62.6	**0.2**	—

Table A1: Comparison of query-level Internal Confidence with answer-level uncertainty methods (Qwen-14B on TriviaQA and SciQ).

propose a training-free method to identify the knowledge boundary of an LLM, which offers a more efficient alternative that can be applied across models and tasks.

C Baseline Details

We adapt existing answer-level methods to quantify the pre-generation uncertainty, e.g., logit-based uncertainty. Given a query (including the prompt) $\mathbf{x} = (x_1, \ldots, x_N)$, we can obtain a probability for each token $P(x_n \mid x_{<n})$ by performing a forward pass. (1) The baseline $\text{Max}(-\log p)$ measures the query's uncertainty by assessing the least likely token in the query [Manakul et al., 2023]. (2) *Predictive Entropy* is defined as the entropy over the entire query token sequence [Malinin and Gales, 2021]:

$$\text{PE}(\mathbf{x}) = -\sum_{n=1}^{N} \log \text{P}(x_n \mid x_{<n}) \tag{A.1}$$

(3) *Min-K Entropy* combines the ideas of $\text{Max}(-\log p)$ and predictive entropy, by selecting the top-K tokens from the query with the minimum token probability [Shi et al., 2024]. (4) *Attentional Entropy* is a modified version of the predictive entropy that considers a weighted sum:

$$\text{AE}(\mathbf{x}) = -\sum_{n=1}^{N} \alpha_n \log \text{P}(x_n \mid x_{<n}), \tag{A.2}$$

where α_n are the attentional weights for tokens x_n. The intuition here is that tokens contribute to the semantic meanings in different ways, such that we should not treat all tokens equally [Duan et al., 2024]. (5) *Perplexity* reflects how uncertain a model is when predicting the next token:

$$\text{PPL} = \exp\left(-\frac{1}{N}\sum \log \text{P}(x_n \mid x_{<n})\right) \tag{A.3}$$

(6) *Internal Semantic Similarity* measures the average similarity among hidden states of different layers $\{\mathbf{h}_N^{(1)}, ..., \mathbf{h}_N^{(L)}\}$, which is inspired by lexical similarity [Fomicheva et al., 2020]. (7) $P(\text{YES})$ is the probability of self-evaluation, as defined in Equation 1. (8) *Internal Confidence (w/ naive avg)* is a simplified variant of our proposed Internal Confidence. The difference is that we compute a naive average to aggregate all scores.

D Additional Experiments

D.1 Calibration Performance

Figure A1 compares the distributions of Internal Confidence scores for known (green) and unknown (blue) queries across three datasets. The results reveal that Internal Confidence tends to assign higher

values to known queries and lower values to unknown queries, which is suitable for distinguishing the two groups. Specifically, on TriviaQA, the separation is mild with noticeable overlap. On SciQ, the known queries concentrate near 1.0, while unknown queries spread toward lower scores, and on GSM8K, the distinction is the clearest, with known queries clustered in the high-confidence region (0.8–0.9) and unknown queries shifted leftward.

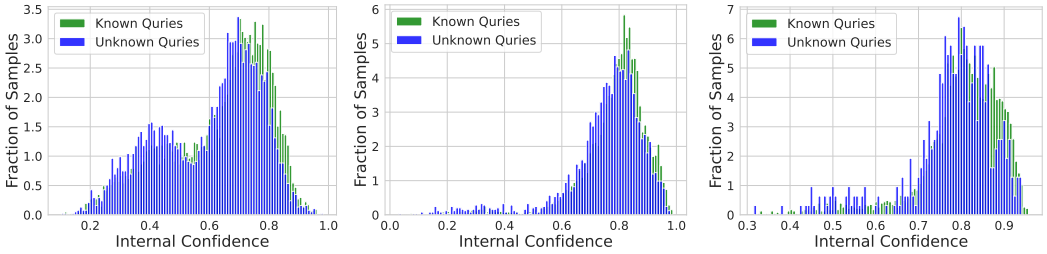
D.2 Internal Confidence does not rely on in-context learning

Figure A2 shows the effect of the number of in-context learning example pairs (k-shot) on model performance across three datasets and models. Here, we randomly select k pairs of positive and negative samples. We plot the AUC as a function of k-shot values from 1 to 5. Overall, Llama-8B and Qwen-14B maintain relatively stable performance with slight improvements as k increases, while Phi-3.8B exhibits more fluctuation, especially on TriviaQA. These results suggest that the benefit of additional in-context examples varies across both models and datasets. Therefore, our Internal Confidence can obtain strong performance even without in-context learning from examples, which can reduce the computational cost.

D.3 Impact of Locality

Figure A3 presents the impact of locality on AUC performance across three datasets (TriviaQA, SciQ, GSM8K) and three models (Phi-3.8B, Llama-8B, Qwen-14B). For Phi-3.8B, AUC improves gradually with increasing locality across all datasets, with TriviaQA exhibiting consistently higher discriminability than SciQ and GSM8K. For Llama-8B, the performance remains fairly stable across different locality values, showing only minor fluctuations, particularly for SciQ and GSM8K. For Qwen-14B, the AUC increases with the locality for all datasets up to a certain point, after which it either plateaus or slightly declines; this trend is most evident for GSM8K.

Locality has a non-trivial effect on the performance of Internal Confidence, and its optimal value varies slightly by model and dataset. Phi-3.8B and Qwen-14B benefit more clearly from tuning locality, while Llama-8B appears more robust to changes. Overall, high locality values often yield competitive or optimal performance.



(a) TriviaQA (b) SciQ (c) GSM8K

Figure A1: Distinguishing between known and unknown queries using Internal Confidence for Phi-3.8B.

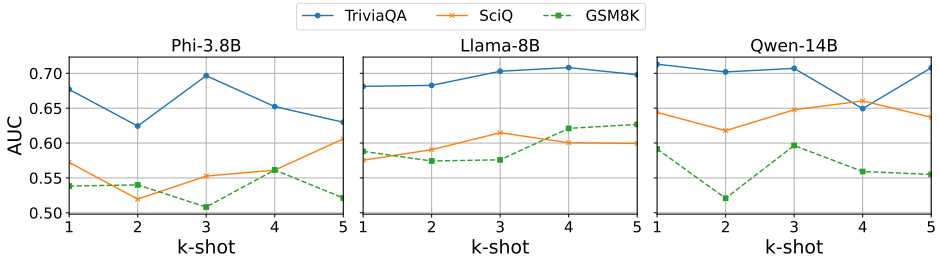


Figure A2: Impact of the number of in-context-learning example pairs on validation set performance.

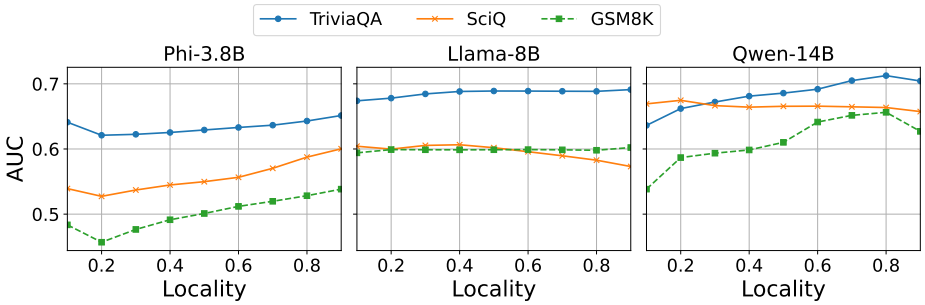


Figure A3: Impact of locality on validation set performance.