UCE: Unified Confidence Estimation Throughout the Generation of LLMs

Anonymous submission

Abstract

Empowering large language models (LLMs) to accurately express confidence in their responses is critical to improve the reliability of LLMs. Existing methods estimate the confidence from a restricted perspective and only for specific token positions, focusing mainly on the estimation of question and entire generated answer. They overlook the continuous monitoring the confidence levels throughout the generation process. Moreover, their methods tend to be task-specific, making it difficult to apply to more general tasks. In this paper, we introduce the **U**nified **C**onfidence **E**stimation (UCE) method, which provides the accurate and continuous confidence estimates throughout the generation process of an LLM. It is also a universal method that offers confidence estimates for any given text sequence. Specifically, we develop a pipeline to construct training data to capture the inherent certainty of LLMs, and design data formats for three different scenarios to improve the generalization capability of LLM confidence estimation. Additionally, we propose the Reverse Confidence Integration strategy, which integrates confidence scores from subsequent text sequences to provide a more accurate and holistic confidence estimation for the current text sequence. In our experiments, we employ two commonly used metrics in confidence estimation and evaluate our method across multiple datasets. The results demonstrate that UCE consistently outperforms existing methods in various confidence estimation tasks. The training dataset and all baselines used in our experiments are available in the GitHub.[1]

Introduction

Large language models (LLMs) typically utilize extensive text corpora for pre-training and subsequently undergo instruction fine-tuning on supervised data (Ouyang et al. 2022; Wei et al. 2021). Reliable estimates of certainty or uncertainty are vital for effective human-machine collaboration, facilitating more logical and well-informed decision-making processes (Tomani and Buettner 2019). Specifically, accurate confidence estimation of LLMs can provide important signals for their generation reliability, safe deployment (Tomani et al. 2024), selective generation (Ren et al. 2022), mitigation of hallucinations (Xiao and Wang 2021) and continuous evolution (Ren et al. 2023).

[1]https://anonymous.4open.science/r/Unify-Confidence-Estimation-7A5F/

The existing LLM confidence estimation methods mainly focus on calculating the confidence scores **for specific token positions from a limited perspective, neglecting a comprehensive estimation of confidence.** These works can generally be categorized into two groups: question-oriented and outcome-oriented confidence estimation. *The question-oriented confidence estimation task instructs LLMs to only respond to questions within their scope and refuse to answer unknown questions* (Zhang et al. 2023; Kadavath et al. 2022), aiming to enhance cognitive self-awareness and determine the knowledge boundaries of LLMs. On the other hand, *the outcome-oriented confidence estimation task requires LLMs to evaluate the quality of their entire generated answers* (Zhang et al. 2024a; Zhao et al. 2024; Kuhn, Gal, and Farquhar 2023; Abbasi-Yadkori et al. 2024). This post-hoc self-evaluation task bolsters the transparency of the LLMs outputs. Differences between them are shown in Figure 1.

However, **it is also necessary to estimate the quality of the intermediate generation steps.** A correct final response does not guarantee the correctness of all preceding steps in the generation process; Similarly, an incorrect final answer does not necessarily mean that all intermediate steps are flawed (Lai et al. 2024). While the accuracy of the final output is crucial, evaluating the correctness and quality of the intermediate steps offers a deeper insight into the the completeness of model's generated answers and its specific week points, which are critical for the continuous refinement and evolution of LLMs. Although a few works attempt to evaluate the generation process of LLMs (Yao et al. 2024; Wang et al. 2024a,b), **they are primarily designed for specific tasks, such as mathematical reasoning tasks, and provide discrete quality estimation scores for each reasoning step.** However, for most tasks, especially open-ended questions, it is challenging to segment the generated answers into multiple separate and manageable steps, making these methods difficult to apply to more general tasks.

To address these limitations, we propose a **U**nified **C**onfidence **E**stimation method (UCE) in this paper to provide a **continuous** and **comprehensive** confidence estimation throughout the entire generation process. UCE is a **universal** method capable of providing accurate confidence estimates for any given text sequence. However, it presents two main challenges in implementing this method. Firstly, ***how***

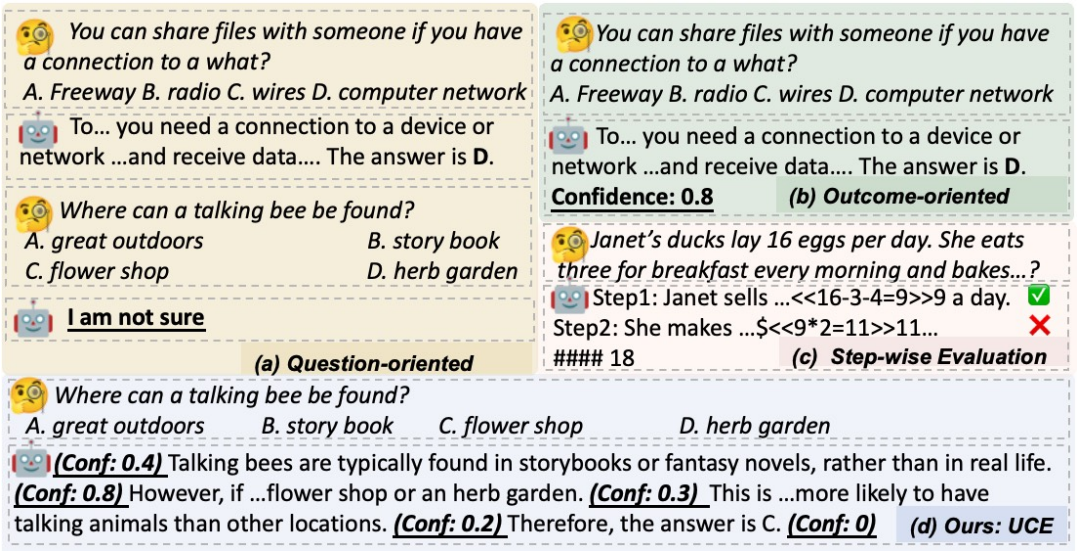

Figure 1: Three tasks of confidence estimation in LLMs and step-wise evaluation in mathematical reasoning task. **(a):** For questions within the model's knowledge boundary, the LLM provides an answer(top left); For questions that surpass its knowledge boundary, the model refuses to respond (bottom left). **(b):** For any given question, the model provides the confidence score alongside an answer. Here, we provide the two of the most commonly used methods. **(c):** Existing step-wise evaluation focuses on the mathematical reasoning task, and provides a evaluation score for each reasoning step. **(d):** It provides the continuous confidence estimates for any given text sequence during the generation process.

to build the training dataset to improve the generalization ability of LLM to provide confidence estimation for any text? In practical scenarios, the text generated by the model is mostly unstructured sequences. To ensure the LLMs offer accurate confidence estimates for any given text, the training dataset need includes standardized and style-free text samples. However, it is difficult to prepare the diverse texts and corresponding confidence scores. Secondly, ***how to provide an accurate and unbiased confidence estimate for the current text?*** Even if the proceeding texts have the same content, LLMs may still generate highly variable subsequent texts (Atil et al. 2024). If we only consider the local confidence estimate of the current text while neglecting the confidence estimate of the subsequent texts, the resulting confidence scores may exhibit systematic bias.

Accordingly, we devise a process of constructing training data to enable LLMs to accurately estimate confidence, and design data formats for three different confidence estimation task scenarios to further improve the model's generalization capability. Additionally, to provide a more accurate and holistic confidence estimation for the current text, we introduce the Reverse Confidence Integration (RCI) strategy during the testing phase, which integrates the confidence estimation derived from the subsequent texts and enables correct overly high or low confidence estimates. In summary, our contributions are as follows:

- We propose a unified confidence estimation method that provides continuous and comprehensive confidence estimates throughout the generation process. At the same time, this method is universally applicable, offering accurate confidence estimates for any given text sequence.
- We provide a detailed data construction process to enhance the confidence expression ability of LLMs and further validate its generalization ability on new tasks.
- In order to avoid generating local confidence estimates and correct the overly high or underestimated estimates, we introduce a novel strategy in the testing phase, Reverse Confidence Integration, by integrating the confidence of the subsequent text to generate a holistic confidence estimate for the current text.
- Extensive experiments demonstrate that our proposed UCE achieves the best confidence estimation performance compared to existing methods on each tasks.

Related Work

Knowledge Boundary Detection of LLMs. While LLMs possess extensive parametric knowledge, they still lack the human-like ability to recognize what they know and what they don't know. To enhance their awareness of knowledge boundaries, efforts have been concentrated on enabling models to decline to answer questions beyond their knowledge scope ("don't know") and provide responses within its scope ("know"). Current works on self-awareness in LLMs can be grouped into two main approaches. One employed the models' internal parameters or structural information to assess their capability to address specific questions (Su et al. 2024; Chen, Vondrick, and Mao 2024). For example, Chen et al. (2024) developed a method involving matrices derived from the model's multiple internal output vectors to calculate eigenvalues to detect errors. Azaria and Mitchell (2023) trained a classifier to identify the correctness of responses based on the hidden layer activation states in the LLM. The

other approach was to evaluate LLM's performance on a set of questions to probe its knowledge limits. R-tuning (Zhang et al. 2023) utilized labeled data to evaluate the correctness of the generated answer and trained the model through supervised fine-tuning.

However, these methods tended to be overly strict and conservative. Faced with uncertain questions, LLMs refused to answer the question rather than attempting to deduce a potential answer from available information. This overly cautious strategy diminished the utility of LLMs. Confidence Expression can alleviate this problem by allowing LLMs to generate likely answers while conveying their real confidence levels for the provided answer.

Confidence Expression in LLMs. In terms of confidence expression in LLMs, existing works have focused on evaluating the certainty or uncertainty of LLMs in generating correct answers to specific questions. One approach was to use carefully designed prompts to guide LLMs to express their confidence level in words along with the generated answers (Zhou, Jurafsky, and Hashimoto 2023; Xiong et al. 2023). Branwen (2020) displayed GPT-3's capability to convey uncertainty on basic questions through few-shot prompts. Lin, Hilton, and Evans (2022) introduced the concept of "verbalized confidence", which directly guided LLMs to output the confidence. Tian et al. (2023a) employed external annotations to instruct LLMs to express uncertainty in words during the answers generation processes. However, it was shown that LLMs exhibit high confidence when prompted to verbalize their confidence (Xiong et al. 2023), and they often struggle to follow complex instructions.

Another line of works focused on leveraging the logit values of specific tokens (e.g. A, B, C, etc) in the generated answer to measure the uncertainty of the entire answer sequence (Robinson, Rytting, and Wingate 2023). Kadavath et al. (2022) proposed probing the self-awareness of LLMs by incorporating a dedicated "Value Head". However, this method faced challenges when applied to general tasks due to its reliance on structured datasets, like multiple-choice questions. Moreover, there has been significant progress in developing metrics to measure the certainty of LLM responses. Kuhn, Gal, and Farquhar (2013) proposed utilizing semantic entropy among multiple sampled answers under the same questions to estimate model's uncertainty. The semantic similarity is quantified using a separated natural language inference classification system (NLI). Zhang et al. (2024b) decomposed LLMs' confidence into two dimensions, including the uncertainty about the question and the fidelity to the answer generated by the LLM.

Overall, current methods usually utilize the inherent capabilities or signals of LLMs to instruct their expression of confidence. These methods rely more on the capabilities of the model itself, targeting tasks with standardized answers. In contrast, in this paper, we consider the ability to express confidence as a meta-capability that requires explicit training within LLMs.

The similar to our work is to evaluate the reasoning steps (Wang et al. 2024a; Lightman et al. 2023) or the generation answers (Cobbe et al. 2021a) by training a reward model. These methods aimed to rank the multiple generated answers and select the best one or construct the step-wise data (Lai et al. 2024). However, they were designed for a particular task such as mathematical reasoning, and provided the discrete evaluation score for the reasoning steps to improve the final reasoning performance. Besides, they overlooked discussing the accuracy of evaluation. In contrast, we focus on exploring a universal method that can provide the accurate and continuous confidence estimates for any given text.

Methods

In this section, we first present the task formalization of LLM's confidence estimation. Subsequently, we introduce a complete process to construct the training dataset and the training method for UCE. Then we introduce the Reverse Confidence Integration strategy used during the test phase.

Task Formalization

Existing LLMs generally generate responses in an autoregressive manner, sequentially predicting the next token based on the preceding sequence. Specifically, they consider a sequence of generated tokens $\{t_1, t_2, \ldots, t_n\}$, where n represents the total number of tokens for the given input x. Each token t_i ($i \in [1, n]$) is generated from the probability distribution $P_i = \mathcal{P}(\cdot|x, t_{<i})$, which is conditioned on the input x and the preceding tokens $t_{<i} = \{t_1, t_2, \ldots, t_{i-1}\}$, spanning the whole vocabulary $\mathcal{V}$.

In this paper, our goal is to empower the LLM to accurately estimate confidence for any given text sequence, especially before, during, and after generation process. We define **the confidence for a text sequence as the likelihood of yielding an appropriate or correct response.** Our method is formalized as follows:

$$Conf_s = p(y = \bar{Y}|s) \tag{1}$$

Here, $Conf_s$ is the confidence estimate for sequence s, which takes the value $[0, 1]$, with larger values denoting a higher degree of certainty. The sequence $y = \{t_1, t_2, \ldots, t_n\}$ represents the complete generated sequence, while $\bar{Y}$ corresponds to the golden answer. The symbol $||$ represents the concatenated operation, while p denotes the probability.

Question-oriented confidence estimation means that the LLM estimates the probability that it can provide a correct answer based on a query, which is represented as $Conf_x = p(y = \bar{Y}|x)$. The outcome-oriented confidence estimation evaluates the quality of the whole generated answer. The value $Conf_y$ is binary, indicating true or false. Intermediate-oriented confidence estimation task aims to provide confidence scores for any given text sequence throughout the generation process. When the input text is a question, it is converted into the question-oriented confidence estimation task; When the input contains both a question and a partial answer, it offers confidence scores to assess the correctness of the current partial answer throughout the generation process; When the input is a complete answer, it shifts accordingly to the outcome-oriented confidence estimation. Therefore, the intermediate-oriented confidence estimation unifies the existing confidence estimation tasks and provide continuous confidence scores throughout the generation process.

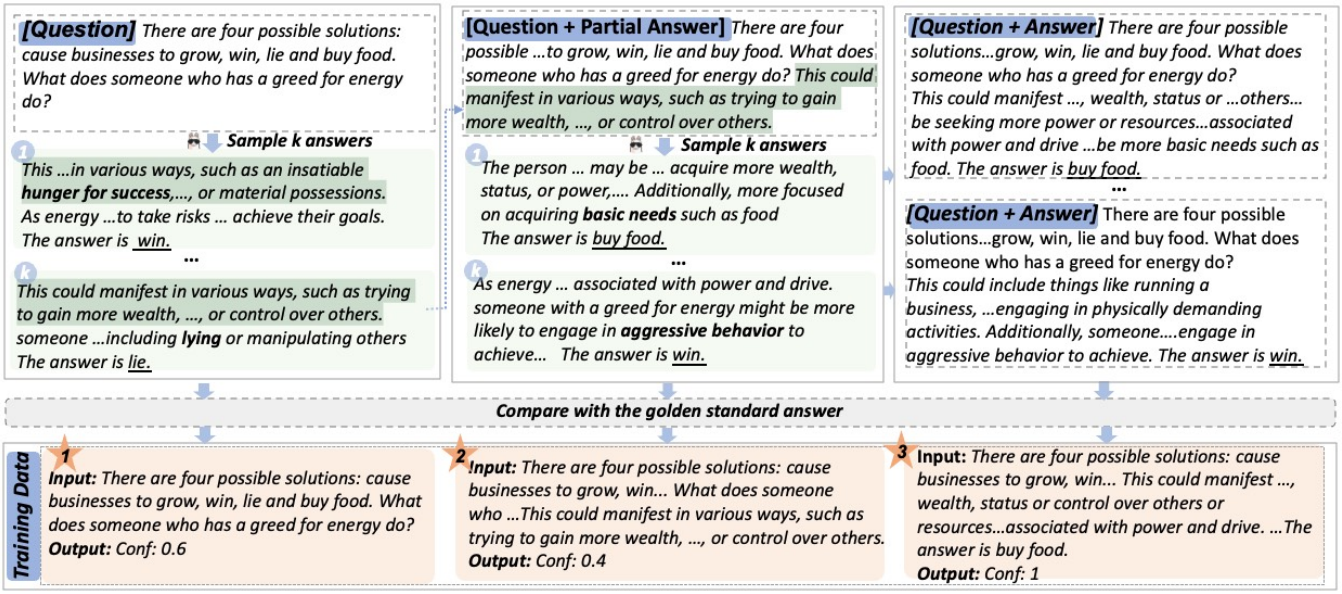

Figure 2: The process of constructing confidence estimation training data. Three data formats are shown in the orange rectangular area. The green highlighted part is the source of the "partial answer".

Unified Confidence Estimation (UCE)

Data Preparation Traditional deep learning approaches for classification fail to capture the model uncertainty. The predictive probabilities provided by the softmax output are frequently misinterpreted as a measure of the model's confidence. However, the model may still be uncertain in its predictions despite producing a high softmax output (Gal and Ghahramani 2016). Therefore, to obtain its inherent real responses, we employ an LLM M to generate k answers $\{A_s^1, A_s^2, \ldots, A_s^k\}$ via random sampling based on the input text sequence s. In our work, the input text sequence s includes three distinct types: *Question*, *Question with Partial Answer* and *Question with Answer*. The value of $Conf_s$ for each text sequence s is decided by the accuracy of k answers generated based on s. Specifically, the confidence score is determined by the proportion of these k answers that are correct or appropriate compared to a standard or golden answer $\bar{Y}$, which is calculated by:

$$Conf_s = \frac{\sum_{i=1}^{k} \mathbf{I}(A_s^i) = \bar{y}_s}{k}, \quad (2)$$

where A_s^i represents the ith sampling answer generated based on sequence s. For the closed questions, $\bar{y}_s$ is defined as the standard answer. $\mathbf{I}$ denotes the indicator function, if the generated answer is consistent with the standard answer, the value is 1, otherwise it is 0. For open-ended questions, the evaluation results can be from stronger LLMs such as GPT-4 serve as the confidence score.

The process of constructing the training data is shown in Figure 2. Initially, for a given question x, the model generates k potential answers. The accuracy of these responses is determined by being compared with a predefined standard answer and utilized as the confidence for question x. According to the value of the final answer, these k responses are classified into m distinct types ($m \leq k$). For each type, randomly select partial answers $(\tilde{A}_x^i)_j$ ($j \in [1, m]$) to ensure a diverse representation of response types. This process is repeated to determine the confidence for $x||(\tilde{A}_x^i)j$.

It is important to note that when extracting fragments from answers A_x^i, the segmentation can be based on structural human-defined elements such as steps, paragraphs or length. Here, based on the answer inferred in the previous step, we divide it into paragraphs to obtain partial answers. To expand the scale of the training dataset for process-oriented confidence, the step of selecting partial answers can be performed for multiple times.

In order to obtain the overall confidence estimation of the entire generated responses, all the answers generated for question x are first classified based on the result value of the final answer. Subsequently, the generated answers are randomly chosen from each type and compared against the standard answer. The confidence score of each accurately aligned answer is set to 1, while the confidence score of an incorrect answer is 0.

Therefore, we obtain a set of tuples like $\langle s, Conf_s \rangle$. We also explore two different training methods to empower the LLMs to estimate confidence for any given text sequence, including the Additional Value Head and Instruction Fine-Tuning (IFT) (Ouyang et al. 2022). With the additional value head, we treat the confidence estimation as either a regression task or a multi-classification task, which facilitates the output of the confidence level at each token position. Conversely, the IFT training method directs the LLM to generate the confidence in a natural language manner. In the experiments, we compared the results of these two training methods in detail. Finally, the UCE employs the IFT training method, and the training data format is shown in the orange

Base Models	Task	Baselines	GSM8K			CSQA			TrivalQA		
			ACC↑	ECE↓	AUROC↑	ACC↑	ECE↓	AUROC↑	ACC↑	ECE↓	AUROC↑
LLaMA2-13B	Question-oriented	*P(IK)*	30.4	14.50	64.8	**69.9**	29.90	59.5	**66.2**	18.72	65.0
		UCE(ours)	**33.6**	**8.99**	**67.3**	65.6	**16.20**	**69.3**	64.8	**15.51**	**68.4**
	Outcome-oriented	First-Prob	30.4	23.35	59.7	62.5	22.39	60.1	63.1	27.64	57.1
		SuC	31.0	28.82	57.3	60.1	27.23	56.7	62.8	23.51	58.2
		Verb	31.0	29.38	56.2	64.3	21.79	58.3	**65.1**	27.15	53.7
		Fidelity	/	/	/	54.5	18.36	67.1	/	/	/
		UCE(ours)	**33.6**	**5.05**	**77.8**	**65.6**	**11.50**	**70.5**	64.8	**12.01**	**76.9**
LLaMA2-7B	Question-oriented	*P(IK)*	**30.7**	16.36	62.8	**64.8**	24.72	57.4	**57.4**	20.93	68.3
		UCE(ours)	30.3	**13.12**	**72.9**	63.7	**15.97**	**69.5**	53.9	**19.18**	**68.9**
	Outcome-oriented	First-Prob	29.7	25.42	58.1	62.1	25.38	57.7	52.8	25.77	55.1
		SuC	29.1	28.73	57.3	63.4	22.79	55.8	52.1	29.34	57.4
		Verb	**30.3**	28.10	56.2	62.5	26.43	55.4	**54.2**	28.61	55.8
		Fidelity	/	/	/	**40.6**	14.09	68.9	/	/	/
		UCE(ours)	**30.3**	**6.54**	**78.9**	63.7	**11.77**	**72.3**	53.9	**15.45**	**76.8**

Table 1: Results of confidence estimates for all baselines across both question-oriented and outcome-oriented task.

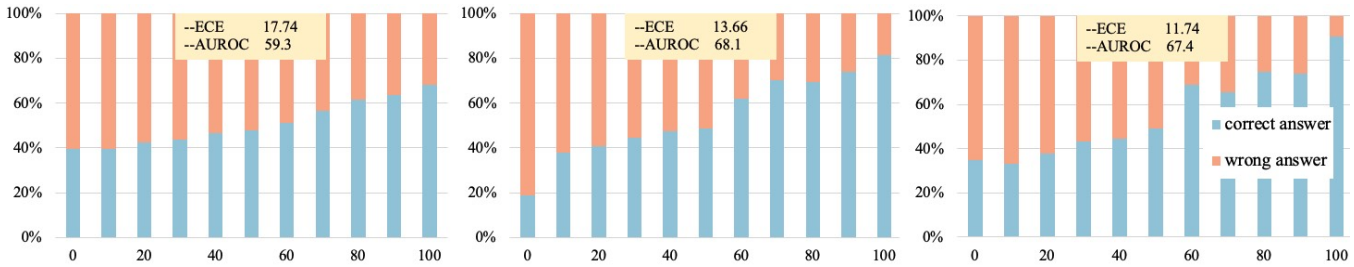

Figure 3: The Zero-shot performance on OpenBookQA dataset. From left to right, the figures show the confidence estimation performance of UCE for the question, partial answer, and complete answer. The horizontal axis represents the confidence estimates (%), and the vertical axis represents the ratio of quantities. The yellow area contains the detailed values of the two metrics.

part of Figure 2.

Reverse Confidence Integration (RSE) After training with the constructed training dataset, we evaluate the performance of the output confidence on a new dataset. LLMs may generate diverse answers even if the input text is the same. To correct either excessively high or low confidence level and reduce the bias in output confidence, we introduce Reverse Confidence Integration strategy. This strategy not only considers the confidence score of the current text but also incorporates the confidence estimates of subsequent text to derive a more holistic confidence for the current text sequence, denoted as $Conf^{'}_{t^i}$. Supposed that the estimated confidence at position i is denoted as $Conf_{t^i}$, where $t^i = \{t_1, t_2, \cdots, t_i\}$, and that multiple answers are generated by sampling at the current position. Therefore, for the text sequence t^i, the adjusted confidence is calculated by:

$$Conf^{'}_{t^i} = \alpha Conf_{t^i} + (1-\alpha)\frac{1}{w}\sum_{j=1}^{w} Conf^{'}_{t^{i+j}}, \qquad (3)$$

where α controls the correction ratio, and w represents the depth of integration. Such back-to-forward deduction strategy obtain the global and accurate confidence estimation for t^i.

Experiments

We conduct experiments to verify the effectiveness of our method, focusing on confidence estimation performance, generalization ability and analyze the effectiveness of the RSE strategy.

Experiment Setting

Dataset. We evaluate the quality of confidence estimation across three datasets including *GSM8K* (Cobbe et al. 2021b), *TrivalQA*(Joshi et al. 2017) and *CommonsenseQA*(CSQA; Talmor et al. 2018).

Models and Baselines. We consider two base models, namely LLaMA2-7B and LLaMa2-13B (Touvron et al. 2023). And the baselines we compared include the following three types:

- **Question-oriented:** *P(IK)*(Kadavath et al. 2022);
- **Outcome-oriented:** *First-Prob* (FirstP; Santurkar et al. 2023), *SuC*(Lin, Hilton, and Evans 2022), *Verbalized Porb* (Verb Tian et al. 2023b), *Fidelity (Zhang et al. 2024a)* ;
- **Step-wise Evaluation:** *Multi-Step* (MP; Xiong et al. 2023), *LECO*(Yao et al. 2024)

Datasets	Process	Metrics	LLaMA2-13B			LLaMA2-7B		
			Multi-Step	LECO	UCE(ours)	Multi-Step	LECO	UCE(ours)
GSM8K	s(1)	ECE	23.51	19.21	**9.31**	24.57	23.71	**12.93**
		AUROC	55.6	60.5	**73.8**	54.4	59.6	**75.3**
	s(n-1)	ECE	22.84	21.37	**8.44**	29.23	25.64	**13.88**
		AUROC	57.3	59.5	**77.7**	54.6	58.4	**76.8**
	AVG	ECE	21.13	19.68	**6.73**	23.18	18.36	**7.21**
		AUROC	57.1	61.1	**78.1**	59.5	63.4	**78.6**
CSQA	s(1)	ECE	24.84	23.87	**18.3**	30.62	26.2	**15.97**
		AUROC	54.6	57.1	**66.2**	51.4	60.2	**69.5**
	s(n-1)	ECE	26.94	25.74	**16.2**	23.43	24.78	**16.79**
		AUROC	53.2	56	**69.3**	54.7	58.9	**69.8**
	AVG	ECE	23.15	21.43	**11.74**	24.49	19.72	**12.83**
		AUROC	58.6	59.6	**71.3**	56	61.7	**72.5**
TrivalQA	s(1)	ECE	22.17	26.84	**14.59**	27.95	32.42	**20.11**
		AUROC	56.1	53.4	**70.8**	60.3	55.7	**73.6**
	s(n-1)	ECE	25.63	27.39	**15.09**	27.42	29.97	**21.07**
		AUROC	56.4	58.3	**74.2**	59	56.1	**73.3**
	AVG	ECE	22.84	25.53	**11.37**	26.73	28.39	**16.12**
		AUROC	57.2	58.1	**76.1**	60.1	57.4	**77.2**

Table 2: Confidence estimates results of partial generated answers of the baseline method during the generation process.

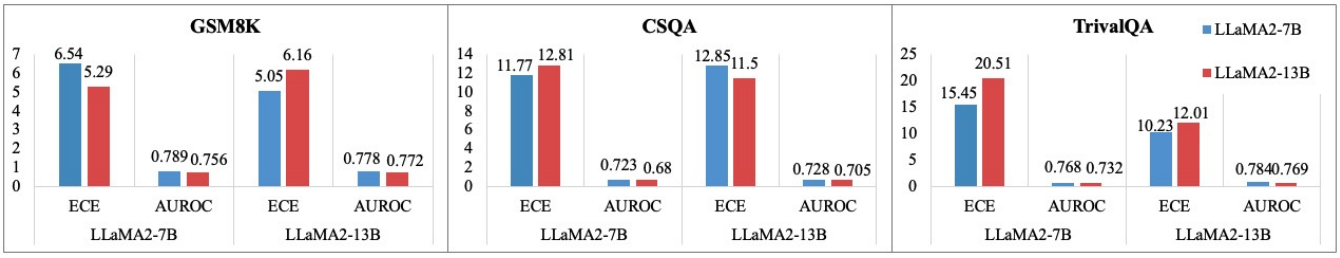

Figure 4: The performance confidence estimation for two base models using training datasets from different sources. The horizontal axis represents the base models

Evaluation Metrics. To evaluate the quality of confidence estimates, we adopt several widely used metrics in confidence estimation: ***Expected Calibration Error (ECE)*** and ***Receiver Operating Characteristic Curve (AUROC)***. ECE evaluates how well a model's confidence estimates aligns with its actual accuracy. AUROC gauges the model's capability to assign higher confidence to correct predictions and lower confidence to incorrect ones, aiming to determine if confidence scores can effectively distinguish between correct and incorrect predictions. Besides, we also use ***Accuracy (ACC)*** to record the accuracy performance of these baselines.

Further details on datasets, models, metrics, and implementations(including all prompts used in this paper, important parameters, and platforms) can be found in Appendix.

Main Results and Analysis.

RQ1: How does UCE perform compared with baselines? We demonstrate that ***base models provide the accurate confidence estimates for any given text sequence on three datasets after using UCE.*** The overall results are shown in Table 1 and Table 2. The results in these two tables are the average values. Particularly, ***our method consistently outperforms all baselines in terms of ECE and AUROC, and shows excellent calibration capability across all datasets.*** Taking the GSM8K dataset as an example, on the answer-oriented confidence estimation task, LLaMA2-13B achieves a lower ECE 5.05%, and the AUROC is as high as 78.9%. At the same time, we observe that although UCE improves the confidence calibration ability through fine-tuning, it does not lead to a decrease in accuracy, showing close accuracy of the outcomes achieved through the prompt engineering method. This is because we conduct the replaying strategy during fine-tuning and mix some general IFT datasets.

From Table 2, we observe that ***UCE delivers the accurate confidence estimates for the partial answer during the generation process.*** Notably, the AUROC values obtained by our method are greater than 70% in most cases, showing a strong performance for accurate identification. In contrast, the AUROC for the other two baselines are always around 60% across these datasets, which is almost close to random guessing. Besides, the outstanding performance on process-oriented confidence estimation task shows that our proposed method UCE can provide the accurate estimates for any given text sequence, which is significantly different from other methods. In the table, $s(1)$ and $s(z-1)$ respec-

Dataset	Base Models	Method	ECE↓	AUROC↑
GSM8K	LLaMA2-7B	w/o	12.93	75.3
		w=2	12.41	74.6
		w=4	11.03	76.1
	LLaMA2-13B	w/o	9.31	73.8
		w=2	9.52	75.9
		w=4	8.94	78.4
CSQA	LLaMA2-7B	w/o	15.97	69.5
		w=2	15.31	69.3
		w=4	12.63	71.7
	LLaMA2-13B	w/o	18.30	66.2
		w=2	15.17	68.6
		w=4	14.28	71.2

Table 3: The performance of RSE on two datasets. w represents the integration depth.

tively represent the first paragraph and the first $z-1$ paragraphs of the generated answer. AVG represents the average confidence estimates for the entire generation process.

To validate the accuracy of the confidence estimates further, we set a confidence threshold. We accept the LLM's generated answers when the output estimates exceed the threshold. We find that for the accepted answers, their real accuracy can be greatly improved, showing a strong calibration ability. For example, on CSQA, LLaMA-13B can achieve 81.8% (+48.2%) accuracy when we set the confidence threshold to 80%. We show more descriptions and experimental results in Appendix.

Generalization Analysis

RQ2: How does UCE perform with zero-shot prompt on new task? To evaluate the generalizability of the UCE method, we test the confidence estimation performance of UCE on OpenBookQA dataset (Mihaylov et al. 2018) using LLaMA2-13B, and the results are shown in 3. We find that UCE exhibits outstanding performance across both the ECE and AUROC confidence metrics. Additionally, there is a robust positive correlation between the model's confidence estimates and the actual accuracy of the answers. Specifically, we observe that higher confidence levels correlated with higher accuracy. It indicates that ***our method possesses noteworthy generalization capabilities and is capable to offer reliable confidence estimates when applied to new tasks.***

RQ3: How does our method perform when trained with datasets from different sources? For the LLaMA2-13B and LLaMA2-7B two base models, we employ two distinct models to construct the training datasets: the model itself or an alternative model. The results are shown in Figure 4. Training with datasets generated from the alternative model achieves confidence calibration performance very close to the obtained using the dataset constructed by the model itself, especially on the GSM8K and CAQA datasets. We guess that it may be related to the used models being from the same family and exhibit significant similarities in their knowledge capabilities. ***It suggests that larger models could effectively instruct smaller models to learn to express the confidence. In addition, leveraging smaller models to construct training datasets may be a cost-efficient alternative.*** In Appendix, we also use two models from different families to explore this phenomenon further.

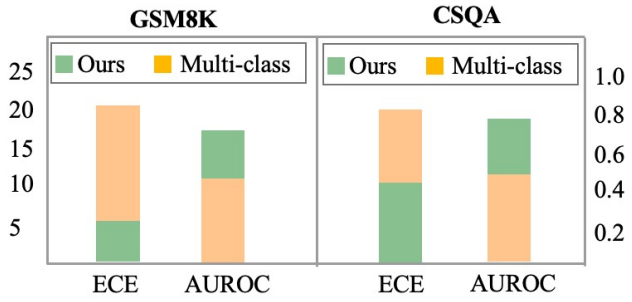

Figure 5: The performance comparison using different training technical. The left side of the vertical axis indicates the value of ECE, and the right side indicates the value of AUROC.

Ablation Analysis

RQ4: How effective is the RSE strategy? To validate the effectiveness of the RSE strategy, we take the confidence score of $s(1)$ as an example and conduct ablation experiments on the GSM8K and CSQA datasets using two base models. The results are shown in Table 3, we find that ***using the RSE method significantly enhances the confidence estimation performance across two metrics*** Moreover, we observed that as the integration depth w increases, the improvement in performance becomes more pronounced.

RQ5: Which training skill is more suitable? On the GSM8K training dataset, we employ two distinct training techniques using the LLaMA2-13B model. One is to add a multi-classification head at the end of the model to output the confidence estimates through classification. The other is the instruction fine-tuning method as we used in the experiment. The outcome confidence estimates results are shown in Figure 5, it suggests that ***under the same data scale, the multi-classification techniques exhibited poor performance in confidence estimation task.***

Conclusion

In this paper, we propose a unified confidence estimation method to enhance LLM's capability to provide accurate and continuous confidence estimates throughout the generation process. We first explain the differences between UCE and existing popular works and then introduce the process of constructing the training dataset. In the testing phase, we propose the RSE strategy that integrates the confidence estimates of subsequent text to generate a holistic confidence estimate for the current text. Moreover, we validate the excellent performance of our proposed method on each type of confidence estimation task through extensive experiments, and further evaluate its generalization ability on other datasets.

References

Abbasi-Yadkori, Y.; Kuzborskij, I.; György, A.; and Szepesv'ari, C. 2024. To Believe or Not to Believe Your LLM. *ArXiv*, abs/2406.02543.

Atil, B.; Chittams, A.; Fu, L.; Ture, F.; Xu, L.; and Baldwin, B. 2024. LLM Stability: A detailed analysis with some surprises. *arXiv preprint arXiv:2408.04667*.

Azaria, A.; and Mitchell, T. 2023. The Internal State of an LLM Knows When It's Lying. In *In Findings of the Association for Computational Linguistics: EMNLP*.

Branwen, G. 2020. Gpt-3 nonfiction- calibration. Technical report, The institution that published. Last accessed on 2022-04-24.

Chen, C.; Liu, K.; Chen, Z.; Gu, Y.; Wu, Y.; Tao, M.; Fu, Z.; and Ye, J. 2024. INSIDE: LLMs' Internal States Retain the Power of Hallucination Detection. *ArXiv*, abs/2402.03744.

Chen, H.; Vondrick, C.; and Mao, C. 2024. SelfIE: Self-Interpretation of Large Language Model Embeddings. *ArXiv*, abs/2403.10949.

Cobbe, K.; Kosaraju, V.; Bavarian, M.; Chen, M.; Jun, H.; Kaiser, L.; Plappert, M.; Tworek, J.; Hilton, J.; Nakano, R.; Hesse, C.; and Schulman, J. 2021a. Training Verifiers to Solve Math Word Problems. *ArXiv*, abs/2110.14168.

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

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

Joshi, M.; Choi, E.; Weld, D. S.; and Zettlemoyer, L. 2017. TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. *ArXiv*, abs/1705.03551.

Kadavath, S.; Conerly, T.; Askell, A.; Henighan, T.; Drain, D.; Perez, E.; Schiefer, N.; Dodds, Z.; DasSarma, N.; Tran-Johnson, E.; Johnston, S.; El-Showk, S.; Jones, A.; Elhage, N.; Hume, T.; Chen, A.; Bai, Y.; Bowman, S.; Fort, S.; Ganguli, D.; Hernandez, D.; Jacobson, J.; Kernion, J.; Kravec, S.; Lovitt, L.; Ndousse, K.; Olsson, C.; Ringer, S.; Amodei, D.; Brown, T. B.; Clark, J.; Joseph, N.; Mann, B.; McCandlish, S.; Olah, C.; and Kaplan, J. 2022. Language Models (Mostly) Know What They Know. *ArXiv*, abs/2207.05221.

Kuhn, L.; Gal, Y.; and Farquhar, S. 2013. Semantic Uncertainty: Linguistic Invariances for Uncertainty Estimation in Natural Language Generation. In *The Eleventh International Conference on Learning Representations*.

Kuhn, L.; Gal, Y.; and Farquhar, S. 2023. Semantic Uncertainty: Linguistic Invariances for Uncertainty Estimation in Natural Language Generation. *ArXiv*, abs/2302.09664.

Lai, X.; Tian, Z.; Chen, Y.; Yang, S.; Peng, X.; and Jia, J. 2024. Step-DPO: Step-wise Preference Optimization for Long-chain Reasoning of LLMs. *ArXiv*, abs/2406.18629.

Lightman, H.; Kosaraju, V.; Burda, Y.; Edwards, H.; Baker, B.; Lee, T.; Leike, J.; Schulman, J.; Sutskever, I.; and Cobbe, K. 2023. Let's Verify Step by Step. arXiv:2305.20050.

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

Mihaylov, T.; Clark, P.; Khot, T.; and Sabharwal, A. 2018. Can a Suit of Armor Conduct Electricity? A New Dataset for Open Book Question Answering. In *Conference on Empirical Methods in Natural Language Processing*.

Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C. L.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; Schulman, J.; Hilton, J.; Kelton, F.; Miller, L. E.; Simens, M.; Askell, A.; Welinder, P.; Christiano, P. F.; Leike, J.; and Lowe, R. J. 2022. Training language models to follow instructions with human feedback. *ArXiv*, abs/2203.02155.

Ren, A. Z.; Dixit, A.; Bodrova, A.; Singh, S.; Tu, S.; Brown, N.; Xu, P.; Takayama, L.; Xia, F.; Varley, J.; Xu, Z.; Sadigh, D.; Zeng, A.; and Majumdar, A. 2023. Robots That Ask For Help: Uncertainty Alignment for Large Language Model Planners. *ArXiv*, abs/2307.01928.

Ren, J. J.; Luo, J.; Zhao, Y.; Krishna, K.; Saleh, M.; Lakshminarayanan, B.; and Liu, P. J. 2022. Out-of-Distribution Detection and Selective Generation for Conditional Language Models. *ArXiv*, abs/2209.15558.

Robinson, J.; Rytting, C. M.; and Wingate, D. 2023. Leveraging Large Language Models for Multiple Choice Question Answering. arXiv:2210.12353.

Santurkar, S.; Durmus, E.; Ladhak, F.; Lee, C.; Liang, P.; and Hashimoto, T. 2023. Whose Opinions Do Language Models Reflect? *ArXiv*, abs/2303.17548.

Su, W.; Wang, C.; Ai, Q.; Yiran, H.; Wu, Z.; Zhou, Y.; and Liu, Y. 2024. Unsupervised Real-Time Hallucination Detection based on the Internal States of Large Language Models. *ArXiv*, abs/2403.06448.

Talmor, A.; Herzig, J.; Lourie, N.; and Berant, J. 2018. Commonsenseqa: A question answering challenge targeting commonsense knowledge. *arXiv preprint arXiv:1811.00937*.

Tian, K.; Mitchell, E.; Zhou, A.; Sharma, A.; Rafailov, R.; Yao, H.; Finn, C.; and Manning, C. D. 2023a. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. *arXiv preprint arXiv:2305.14975*.

Tian, K.; Mitchell, E.; Zhou, A.; Sharma, A.; Rafailov, R.; Yao, H.; Finn, C.; and Manning, C. D. 2023b. Just Ask for Calibration: Strategies for Eliciting Calibrated Confidence Scores from Language Models Fine-Tuned with Human Feedback. *ArXiv*, abs/2305.14975.

Tomani, C.; and Buettner, F. 2019. Towards Trustworthy Predictions from Deep Neural Networks with Fast Adversarial Calibration. *ArXiv*, abs/2012.10923.

Tomani, C.; Chaudhuri, K.; Evtimov, I.; Cremers, D.; and Ibrahim, M. 2024. Uncertainty-Based Abstention in LLMs Improves Safety and Reduces Hallucinations. *ArXiv*, abs/2404.10960.

Touvron, H.; Martin, L.; Stone, K. R.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; Bikel, D. M.; Blecher, L.; Ferrer, C. C.; Chen, M.; Cucurull, G.; Esiobu, D.; Fernandes, J.; Fu, J.; Fu, W.; Fuller, B.; Gao, C.; Goswami, V.; Goyal, N.; Hartshorn, A. S.; Hosseini, S.; Hou, R.; Inan, H.; Kardas, M.; Kerkez, V.; Khabsa, M.; Kloumann, I. M.; Korenev, A. V.; Koura, P. S.; Lachaux, M.-A.; Lavril, T.; Lee, J.; Liskovich, D.; Lu, Y.; Mao, Y.; Martinet, X.; Mihaylov, T.; Mishra, P.; Molybog, I.; Nie, Y.; Poulton, A.; Reizenstein, J.; Rungta, R.; Saladi, K.; Schelten, A.; Silva, R.; Smith, E. M.; Subramanian, R.; Tan, X.; Tang, B.; Taylor, R.; Williams, A.; Kuan, J. X.; Xu, P.; Yan, Z.; Zarov, I.; Zhang, Y.; Fan, A.; Kambadur, M.; Narang, S.; Rodriguez, A.; Stojnic, R.; Edunov, S.; and Scialom, T. 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models. *ArXiv*, abs/2307.09288.

Wang, P.; Li, L.; Shao, Z.; Xu, R. X.; Dai, D.; Li, Y.; Chen, D.; Wu, Y.; and Sui, Z. 2024a. Math-Shepherd: Verify and Reinforce LLMs Step-by-step without Human Annotations. arXiv:2312.08935.

Wang, Z.; Li, Y.; Wu, Y.; Luo, L.; Hou, L.; Yu, H.; and Shang, J. 2024b. Multi-step problem solving through a verifier: An empirical analysis on model-induced process supervision. *arXiv preprint arXiv:2402.02658*.

Wei, J.; Bosma, M.; Zhao, V.; Guu, K.; Yu, A. W.; Lester, B.; Du, N.; Dai, A. M.; and Le, Q. V. 2021. Finetuned Language Models Are Zero-Shot Learners. *ArXiv*, abs/2109.01652.

Xiao, Y.; and Wang, W. Y. 2021. On Hallucination and Predictive Uncertainty in Conditional Language Generation. *ArXiv*, abs/2103.15025.

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

Yao, Y.; Wu, H.; Guo, Z.; Zhou, B.; Gao, J.; Luo, S.; Hou, H.; Fu, X.; and Song, L. 2024. Learning From Correctness Without Prompting Makes LLM Efficient Reasoner. *ArXiv*, abs/2403.19094.

Zhang, H.; Diao, S.; Lin, Y.; Fung, Y. R.; Lian, Q.; Wang, X.; Chen, Y.; Ji, H.; and Zhang, T. 2023. R-Tuning: Instructing Large Language Models to Say 'I Don't Know'. In *North American Chapter of the Association for Computational Linguistics*.

Zhang, M.; Huang, M.; Shi, R.; Guo, L.; Peng, C.; Yan, P.; Zhou, Y.; and Qiu, X. 2024a. Calibrating the Confidence of Large Language Models by Eliciting Fidelity. *ArXiv*, abs/2404.02655.

Zhang, M.; Huang, M.; Shi, R.; Guo, L.; Peng, C.; Yan, P.; Zhou, Y.; and Qiu, X. 2024b. Calibrating the Confidence of Large Language Models by Eliciting Fidelity. arXiv:2404.02655.

Zhao, X.; Zhang, H.; Pan, X.; Yao, W.; Yu, D.; Wu, T.; and Chen, J. 2024. Fact-and-Reflection (FaR) Improves Confidence Calibration of Large Language Models. *ArXiv*, abs/2402.17124.

Zhou, K.; Jurafsky, D.; and Hashimoto, T. 2023. Navigating the grey area: How expressions of uncertainty and overconfidence affect language models. *arXiv preprint arXiv:2302.13439*.

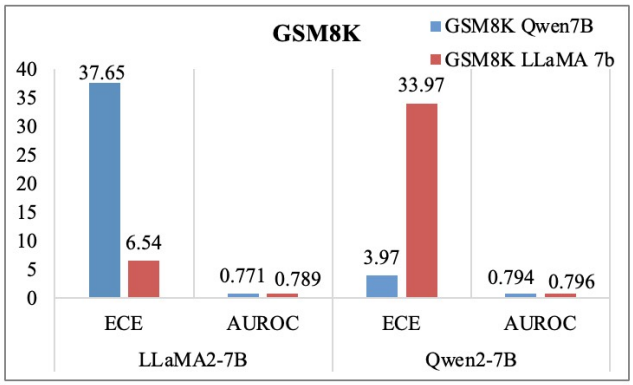

Figure 1: On GSM8K dataset, the performance confidence estimation for the two different families models using datasets from different sources.The horizontal axis represents the base models.

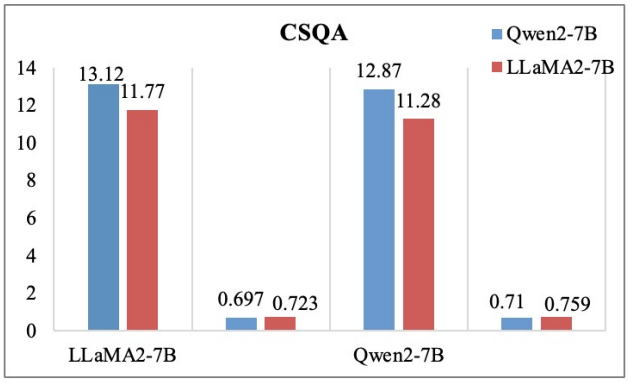

Figure 2: On CSQA dataset, the performance confidence estimation for the two different families models using datasets from different sources.The horizontal axis represents the base models.

Appendix

Baselines. We introduce each method in the baseline, and the prompts used are shown in the Table 2.

- **P(IK).** It trains a logistic regression with the additional value "head" added to the model to output the confidence estimated.
- **First-Prob.** It uses the logits of the first token of LLM's generated answer as the confidence estimate.
- **SuC.** It first clusters the sub-questions, and use the same confidence estimate for questions in the same cluster.
- **Verb.** It is a prompt-based method. It designs the prompts to guide the model to output its confidence score alongside with the generated answer.
- **Fidelity.** For MCQA, it decomposes the LLM confidence into the *Uncertainty* about the question and the *Fidelity* to the answer generated by LLMs.
- **LECO.** It also proposes leveraging logits to estimate step confidence. Besides, it further designs three logit-based scores that comprehensively evaluate confidence from both intra- and inter-step perspectives.
- **Multi-Step.** It also uses prompts to guide the model to output the process confidence and takes the average as the final result.

Important Parameters Settings. During fine-tuning, we employ the AdamW optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.5$. The initial learning rate is set to 1e-4, with the warmup phase of 300 steps. All experiments are conducted on the workstations of NVIDIA A800 PCIe with 80GB memory and the environment of Ubuntu 20.04.6 LTS and torch 2.0.1.

RQ5: Is the confidence estimates really reliable? In order to validate the ability of confidence estimates provided by UCE in verifying the correctness of the generated answer, we set a confidence threshold. Only when the output confidence estimates exceeds this threshold, we accept the corresponding answer. **Compared with unconditionally accepting the output results of the LLM, the accuracy of the model has been significantly improved after introducing output confidence.** The experimental results are shown in the Table 1.

Dataset	Base Models	ACC-before	ACC-after
GSM8K	LLaMA2-7B	30.3	58.8 (+28.5)
	LLaMA2-13B	33.6	78.3 (+44.7)
CSQA	LLaMA2-7B	63.7	79.9 (+16.2)
	LLaMA2-13B	65.6	81.8 (+16.2)
TrivalQA	LLaMA2-7B	53.9	70.3 (+16.4)
	LLaMA2-13B	64.8	80.7 (+15.9)

Table 1: Comparison of the model's accuracy performance across three datasets with a set confidence threshold of 80%.

RQ6: How does our method perform when trained using datasets from different model families? Different from RQ3, we use two base models, including Qwen2-7B and LLaMA2-7B, which are from different model families. The results are show in Figure 1 and Figur 2. We find that there are two different phenomena on different datasets. On the GSM8K dataset, compared with using the model itself to construct training data, the confidence training data constructed with the help of other models performed poorly, especially in the ECE value, where the difference was particularly significant. On the CSQA dataset, the performance difference between the two methods is small. This may be because there is a large difference in the accuracy of Qwen2-7B and LLaMA2-7B on the GSM8K dataset, which makes it impossible to effectively migrate the confidence training data constructed by these two models to each other. We can conclude that **if the performance of two models on a task is close, the confidence training data constructed using one of the models can be effectively used in the training stage of the other model.**

RQ7: How does our method perform on highly open questions? In order to test the generalization ability of UCE on highly open questions, we randomly select 300 single-round English open question-answering data on

Sharegpt [1], and use LLaMA2-7B to provide confidence estimates, and compared the output confidence with the evaluation score of the generated answers using GPT4 to calculate ECE. We find that for highly open questions, our proposed method achieved a higher ECE value of 65.66. This is also in line with our expectations. This is because we did not use GPT4's evaluation to assist in constructing training data, resulting in a large difference between the confidence provided by the model and the GPT4 scoring results.

Limitations Although our proposed method UCE can provide accurate confidence estimates for any given text and we have proven its effectiveness in various tasks, including open-ended text generation, it faces similar challenges when applying UCE to provide confidence estimates for highly open-ended problems as all existing confidence estimation methods. For example, when asked questions such as "*How to stay healthy?*", due to the lack of specific constraints in task instructions, such as perspective constraints or word limit, the model generates multiple different responses. Therefore, to construct the training dataset or test the performance on such task, there is no other more effective method of answer evaluation except to ask more powerful models such as GPT-4 for evaluation. In the future, we will further explore confidence estimation methods for such highly open-ended questions.

[1] https://huggingface.co/datasets/OpenGVLab/ShareGPT-4o

Method	**Prompt**
Verb	Read the question, analyze step by step, provide your answer and your confidence in this answer. Use the following format to answer: "Explanation: [insert step-by-step analysis here] Answer: [ONLY the option letter; not a complete sentence], Confidence (0-100):[Your confidence level, please only include the numerical number in the range of 0-100]%" Please refer to the example I have given: <example> {few-shot} </example> Question: {question} Now, please answer this question and provide your confidence level. Let's think it step by step.
Multi-step	Read the question, break down the problem into K steps, think step by step, give your confidence in each step, and then derive your final answer and your confidence in this answer. Note: The confidence indicates how likely you think your answer is true. Use the following format to answer: Step 1: [Your reasoning], Confidence: [ONLY the confidence value that this step is correct]% Step K: [Your reasoning], Confidence: [ONLY the confidence value that this step is correct]% Final Answer: [ONLY the answertype; not a complete sentence] Overall Confidence(0-100): [Your confidence value]% Please refer to the example I have given: <example> {few-shot} </example> Question: {question} Now, please answer this question and provide your confidence level. Let's think it step by step.
UCE(ours)	Below is a question and some steps: Question: {question} {steps} Please give your confidence.

Table 2: The prompts used in the baseline method.