Confidence Introspection: A Self-reflection Method for Reliable and Helpful Large Language Models

Anonymous ACL submission

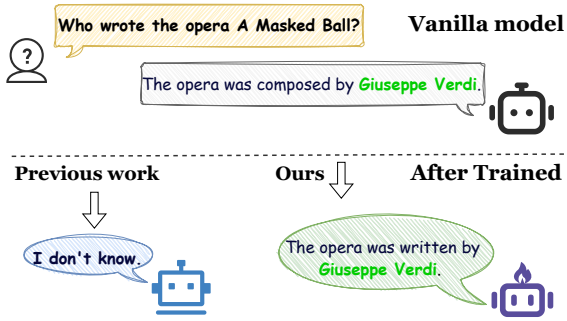
Abstract

Large Language Models (LLMs) suffer from factual hallucinations, meaning the LLMs confidently provide responses that are inconsistent with reality. Previous works designed various calibration methods to mitigate hallucination, but these methods often led the LLM to refuse to answer questions. Inspired by the process of human introspection, we propose Confidence Introspection Training that enables LLMs to accurately express self-confidence and prevent it being overly conservative. The method formalized by first paraphrasing questions and sampling responses to estimate confidence of the LLM. Then the LLM obtains the ability to distinguish among which question is *known*, *uncertain* and *unknown* from learning data generated by itself. Experimental results demonstrate that our method effectively enhances the reliability of the LLM by accurately expressing its confidence level and maintaining the helpfulness of vanilla model in providing useful information [1].

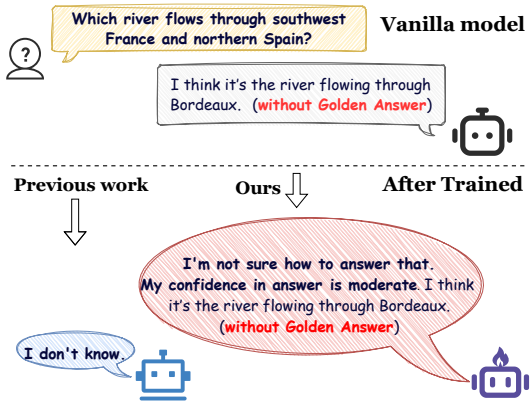
1 Introduction

Reliability is the ability of LLMs to accurately express the confidence level assigned to a prediction (Mahaut et al., 2024). However, factual hallucinations, which lead LLMs to provide responses that contains nonfactual facts, greatly undermine the LLM's reliability.(Perez et al., 2022; Huang et al., 2023; Cheng et al., 2023; Zhang et al., 2024). Confidence calibration is the method that helps enhance reliability of LLMs and mitigate factual hallucinations (Tian et al., 2023).

Previous studies (Cheng et al., 2024) attempted to enhance reliability by classifying questions based on whether the response of the LLM is correct, and trained the model to provide a simple but uniform response like "*I don't know*" when faced with questions it answers incorrectly. These methods are more user-friendly than traditional methods based on token probability (Kuhn et al., 2023). However, these methods also cause the model to become overly conservative, tending to refuse to answer when faced with new questions. As shown in Figure 1(a), the vanilla model can answer correctly while the LLM trained by previous work refuses to respond. Cheng et al. (2024) has made efforts to mitigate this through preference-aware optimization but the effects are limited. They only



(a) Our method answers correctly if the vanilla model knows the answer



(b) Our method provides information closely related to golden answer "Garonne River".

Figure 1: Our method compared to previous work.

[1] Our model and data will be made public after the paper is accepted.

considered reliability, aiming to avoid misleading users, but overlooked **helpfulness** (Bai et al., 2022), which refers to the fundamental characteristic of LLMs to provide information relevant to the questions. This motivates us to seek a confidence calibration method that preserves the helpfulness of vanilla models while ensuring reliability.

Introspection (or self-reflection) (Myers, 1986) is a process of internal communication with oneself, offering clear insights and shaping actions in future situations. Inspired by introspection, we introduce **Confidence Introspection Training, CIT**, which enables the LLM to learn from its own responses, gain insight into its confidence across different questions and retain the helpfulness of vanilla models. As shown in Figure 1(b), our method makes the LLM to express its confidence using certain prefix while providing correct useful information closely related to golden answer. Specifically, we first paraphrase questions and sample responses to estimate the confidence. Then, the LLM obtains the ability to estimate the confidence level of questions and to classify them as *known*, *unknown*, and *uncertain* by trained on the Confidence Introspection Dataset, which is composed of self-generated data. During this process, the LLM can calibrate its confidence estimates for questions while maintaining its ability to provide useful information in its responses. Experimental results demonstrate that, without injecting any external knowledge, the LLM can be tuned to accurately express its confidence level and provide useful information. Compared to previous work, our method enhances the LLM's reliability and maintains the helpfulness of vanilla models, avoiding an overly conservative response tendency. In this paper, our contributions are:

- We propose Confidence Introspection Training, which allows the LLM to generate responses while expressing confidence levels using specific prefixes. Through learning from self-generated data (Confidence Introspection Dataset), the LLM obtains the ability to accurately express confidence and retains the inherent helpfulness of vanilla models.
- Experimental results demonstrate that our method significantly improves accuracy in expressing confidence compared to previous methods, while providing useful information without becoming overly conservative. Moreover, it also performs well on out-of-domain data.
- Ablation studies indicate that our method is applicable to various models and confidence classifications, demonstrating strong generalizability.

2 Confidence Introspection Training

2.1 Confidence Estimation of LLMs

Assessing the LLM's confidence in certain questions is the first step of confidence introspection. Inspired by Kadavath et al. (2022) and Mahaut et al. (2024), we first paraphrase the question and then sample five responses for the original question and another five for the paraphrased question. We calculate the number of correct responses among these ten responses. The method for determining correctness is Exact Match (Wang et al., 2023a), which considers a response correct if it contains the golden answer. We define the total number of correct responses as **CRes**, which we consider to represent the confidence estimation of a specific question. Figure 2 shows the distribution of CRes across questions in the TriviaQA and Natural Questions datasets, while Table 6 in Appendix presents the specific percentage for each CRes.

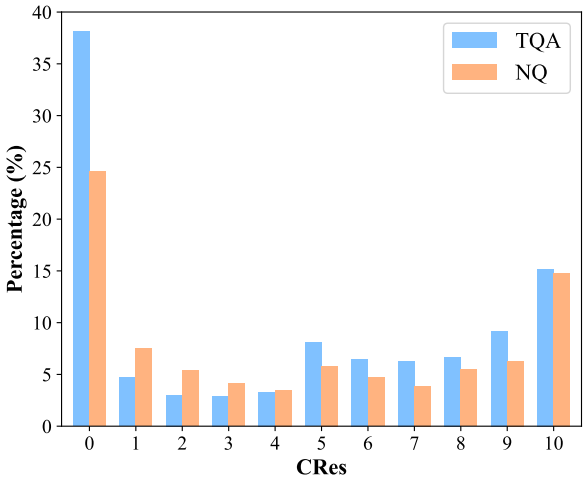


Figure 2: The figure illustrates the distribution of model confidence across questions in TriviaQA (TQA) and Natural Questions (NQ).

We observe that data with CRes ranging from 1 to 9 constitute a significant proportion of the total data (TQA: 46.7%, NQ: 60.7%). Moreover, CRes values between 0 and 10 indicate that LLMs sometimes answer correctly and sometimes incorrectly. Within this CRes range, The understanding of the question by the LLM is ambiguous. Therefore, the LLM cannot always output the correct answer.

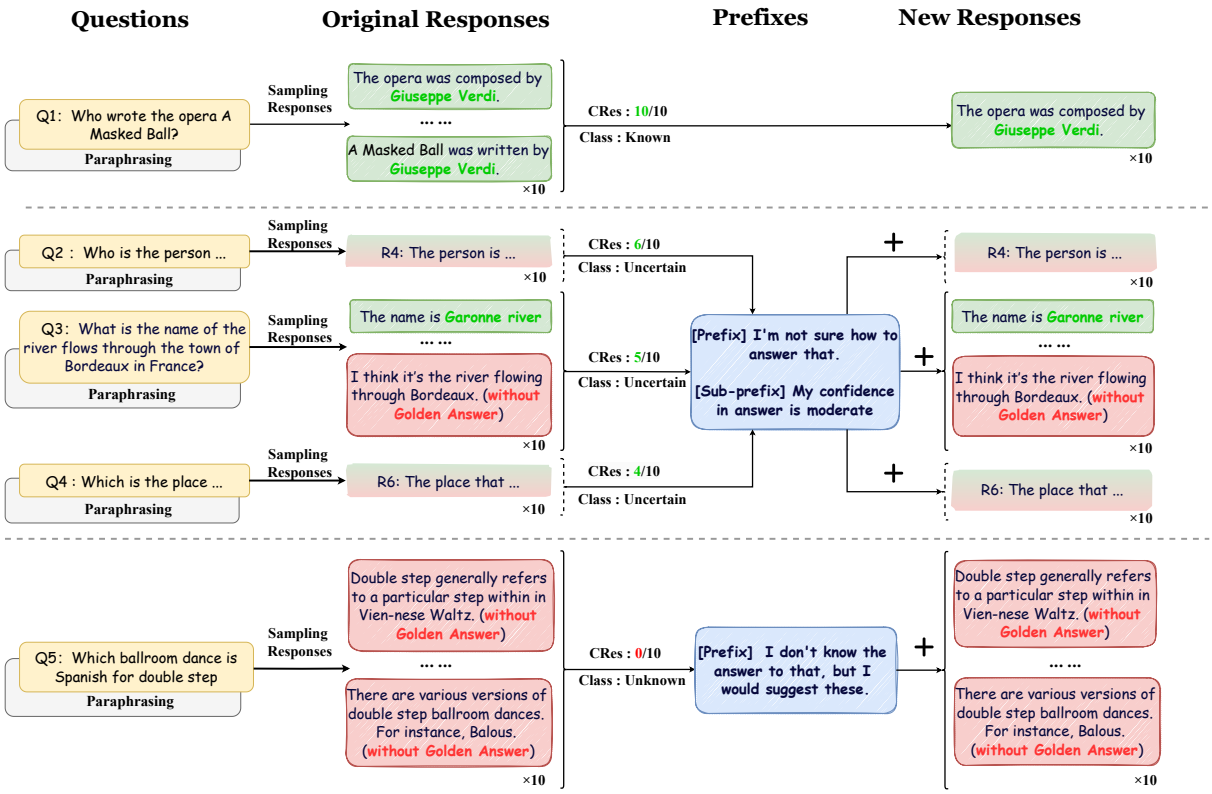


Figure 3: We classify the question based on the CRes as *known*, *uncertain* and *unknown*. We add certain prefixes to responses for *uncertain* and *unknown* questions, forming the expected output data of CIs dataset. In the figure, within the *uncertain* class, we use questions with a CRes between 4 to 6 as example. The data construction process for CRes in 1 to 3 and 7 to 9 is the same.

2.2 Confidence level and prefixes

Due to errors introduced by paraphrasing and sampling, the LLM cannot be directly calibrated using the CRes values. Therefore, it is necessary to classify questions based on their CRes values, assigning each class a unified confidence level. The confidence level can then be used to calibrate the LLM.

Previous works (Cheng et al., 2024; Xu et al., 2024) group all questions that the LLM cannot answer entirely correctly, labeling them as *unknown* questions. By analogy, this is equivalent to considering all questions with CRes below a certain value (often ten) as *unknown* questions. However, they overlook that the LLM has already possesses relevant knowledge for the questions and has the potential to provide correct answers. Worse still, these questions constitute a significant proportion of the overall data (e.g., 60.7% in TQA). Previous works trained the LLM to respond only with *I don't know* when encountering these questions, which causes the LLM to habitually refuse to answer most questions, becoming overly conservative.

Considering the limitations of previous work, We classify questions by CRes into three classes, each corresponding to a distinct confidence level: **known**, **uncertain** and **unknown**, as Equation 1.

$$Class = \begin{cases} \textbf{Unknown}, & \text{CRes} = 0 \\ \textbf{Uncertain}, & 1 \leq \text{CRes} \leq 9 \\ \textbf{known}, & \text{CRes} = 10 \end{cases} \quad (1)$$

The introduction of the *uncertain* class allows the LLM to provide different responses from the *unknown* question, preventing the LLM from always saying *I don't know* and refusing to answer most questions. Moreover, this ensures higher reliability for *unknown* questions. The LLM must answer incorrectly all five times before and after paraphrasing for the question to be considered as *unknown*. At this point, it is likely that the LLM lacks all relevant knowledge of the question, making it reasonable to conclude that the LLM does not know the answer at all.

Prefixes of confidence level Building on the three confidence levels, we make the LLM to express confidence level of question through specific

prefixes before outputting answer. These statements, indicating specific confidence levels, are referred to as prefixes. For example, this is the expected output of our method to an *unknown* question:

$$R_{nk} : \{\textbf{Prefix}\} + \{\textbf{Response}\}$$

The "Response" in expected output refers to the answer of vanilla model to the question. Examples of prefixes of *unknown* questions can be found in Figure 3. Both *unknown* and *uncertain* questions have corresponding prefixes, as detailed in Appendix A.1.

It is worth noting that the prefixes are all generated by the LLM itself. The LLM is prompted to output statements conveying uncertainty or the unknown, with three prefixes sampled for each of the *unknown* and *uncertain* questions. For *known* questions, no prefix is added and the LLM can respond as usual.

Detailed levels and sub-prefix In addition to prefixes, we hope that the LLM can express confidence levels in a more fine-grained manner for *uncertain* questions, as this class includes questions corresponding to nine different CRes. Considering sampling errors, we further subdivide uncertain questions into three sub-classes based on CRes, with each sub-class assigned a specific sub-prefix to indicate the corresponding confidence level (CRes range), as shown in Table 1.

CRes	Sub-Prefixes
$1 \sim 3$	My confidence in answer is **low**.
$4 \sim 6$	My confidence in answer is **moderate**.
$7 \sim 9$	My confidence in answer is **high**.

Table 1: Sub-prefixes of Confidence levels of *uncertain* question.

Therefore, for an *uncertain* question, the LLM is expected to response like:

$$R_{uc} : \{\textbf{Prefix}\} + \{\textbf{Sub-Prefix}\} + \{\textbf{Response}\}$$

2.3 CIs Dataset Construction

We annotated questions from TriviaQA (TQA) (Joshi et al., 2017), Natural Questions (NQ) (Kwiatkowski et al., 2019) and GSM8K (Cobbe et al., 2021) for a specific LLM, forming our Confidence Introspection Dataset (CIs) . Specifically, we calculate CRes using the method in Section 2.1, adding specific prefixes and sub-prefixes to responses to form expected outputs that reflect the model's confidence level. Overall data processing and expected response format is shown in Figure 3.

Then, we divide the data into train dev and test sets (see Table 7 in Appendix). The train and dev sets are from TQA training set, while the test set is from both the TQA and NQ. Data from NQ and GSM8K serves as out-of-domain data to assess the generalization ability of methods. The training, dev, and test sets contain as close to an equal number of questions from each class as possible (*known*, *unknown*, three confidence level of *uncertain*). More details of CIs dataset are shown in the Appendix C.

Notably, as mentioned in Section 2.2, all training data and prefixes comes from the LLM's own generation, which reduces changes in the LLM's responses to the same question, minimizing the impact on the LLM.

2.4 Training Strategy

Based on the CIs dataset, we perform our Confidence Introspection Training, **CIT**, to train the LLM to distinguish between different classes of questions (*known*, *unknown*, and *uncertain* with three confidence levels).

Compared to previous methods (Cheng et al., 2024; Xu et al., 2024) that typically classify questions as *known* or *unknown*, we introduce a new uncertain class with specific prefixes, offering the LLM more ways to respond and providing users with finer-grained insights into the LLM's confidence. Loss of CIT is shown in Equation 2.

$$L = -\frac{1}{N}\sum_{1}^{N}\begin{cases} \log(R|Q^T) & T = \textbf{kn} \\ \log(P^T + R|Q^T) & T = \textbf{uk} \\ \log(P^T + C^T + R|Q^T) & T = \textbf{uc} \end{cases} \tag{2}$$

Here, P denotes the prefix, R and Q^T represent the response and different confidence level of questions, C^T represents the sub-prefix in Table 1. When the question type is *uncertain* (**uc**) or *unknown* (**uk**), the LLM needs to generate the prefix before responses. When the type is *known* (**kn**), no prefix is required.

3 Experiments

We treat the identification of confidence level (*known*, *uncertain*, three level of *unknown*) as a

Methods	TriviaQA			Natural Questions			GSM8K		
	ACC	EM-K	EM-U	ACC	EM-K	EM-U	ACC	EM-K	EM-U
I-Prompt	22.72	61.18	39.31	21.28	55.51	32.97	6.02	35.44	8.51
S-Token	24.36	68.58	54.03	22.56	57.27	52.27	20.76	24.71	37.95
SKR_{cls}	24.42	70.33	50.14	25.16	61.44	51.70	21.55	31.25	18.46
Pf-SFT	25.56	72.02	54.27	22.96	55.29	48.61	14.90	31.47	31.64
IDK-DPO	27.00	68.81	-	22.40	25.00	-	22.66	12.90	-
CIT	**29.52**	**86.78**	**56.15**	**25.96**	**62.79**	**52.99**	**25.19**	**64.71**	**39.87**

Table 2: The experimental results for our methods CIT and other baselines. The results demonstrate that our method achieves the best results on almost all metrics. Especially, the results on the NQ and GSM8K data confirm that our method is more robust.

five-classification task. Most existing baselines can only perform binary classification (*known* and *unknown*), so we modify them to adapt to our five-classification task.

3.1 Baselines

Introspection Prompting (I-Prompt) We utilize natural language instructions to prompt the vanilla model to answer questions and inform users of its confidence. Please refer to the Appendix A.2 for the instructions.

Surrogate-Token (S-Token) Mahaut et al. (2024) prompted the LLM to provide specific tokens, and the probabilities assigned to those tokens are then used to determine the confidence level of the questions.

IDK-DPO Cheng et al. (2024) employs Direct Preference Optimization (Rafailov et al., 2023) to train the LLM to say *I don't know*. We train the LLM to output certain prefix and confidence level for *uncertain* and *unknown* questions, while outputting responses only for *known* questions.

SKR_{cls} Wang et al. (2023b) trained a BERT (Devlin et al., 2019) to classify questions into different classes and use a vanilla LLM to generate answers.

Pf-SFT Liu et al. (2023) introduced preference information in the SFT method by marking the correct and incorrect outputs with "GOOD" and "BAD" labels. We use their method to distinguish between different classes of questions.

3.2 Evaluation Metrics

We adapt "Truthful" metric from Cheng et al. (2024) and transform it into Introspection Accuracy to evaluate reliability, and introduce the Exact Match (EM) used in QA tasks to evaluate helpfulness.

Introspection Accuracy (ACC) This is the accuracy of this five-class classification task. We determine the question's classification based on whether the certain prefix and sub-prefix appear in the response. If no prefix is present, the class is considered *known* [2].

Generation Exact Match (EM) Exact Match measures whether the response contains the golden answer, which is an important aspect of helpfulness [3]. Since responses to *known* or *uncertain* questions are more likely to be utilized by users, Exact Match only evaluates the responses of these two classes, denoted as **EM-K** and **EM-U**.

3.3 Implementation Details

We use the Mistral-7B-Instruct (Jiang et al., 2023) for our CIT method and other baselines. The learning rate is set to 2×10^{-5}, with 3 training epochs on. We conduct all experiments on Nvidia A800 GPUs. For more details, please refer to the Appendix C.

3.4 Main Results

Reliability As shown in Table 2, CIT method demonstrates significant improvement in ACC compared to baselines. On the TQA data, CIT outperforms other methods and achieves performance comparable to that of SKR_{cls}, which was specifically trained for five-classification task. The results

[2]For the vanilla model (I-Prompt), we relax the evaluation criteria slightly by adding more prefixes, shown in Appendix A.3

[3]For GSM8K, we take the last number in the response to compare with the golden answer.

on the out-of-domain (NQ and GSM8K) data indicate that the CIT method is more robust and has better generalization ability. The LLM has gained the ability to independently judge the class of questions, rather than memorizing the training data.

Helpfulness Providing correct responses is an important aspect of helpfulness. CIT method shows improvement in EM compared to other methods.This indicates that CIT improves generation accuracy based on higher classification accuracy. Training on self-generated data effectively avoids disrupting the model's existing knowledge during the SFT process and can even strengthen the model's understanding of own knowledge, thereby leading to a improvement in generation accuracy. The results on the out-of-domain NQ data demonstrates that our method is very robust. The low EM-K and EM-U for I-prompt is due to the poor classification ability, which incorrectly classify *unknown* questions (i.e., the LLM cannot answer correctly) into the *known* and *uncertain* class.

4 Analysis

4.1 Ablation Study

We analyze the impact of number of confidence level classes, the size and type of LLMs on the classification ability (Introspection Accuracy).

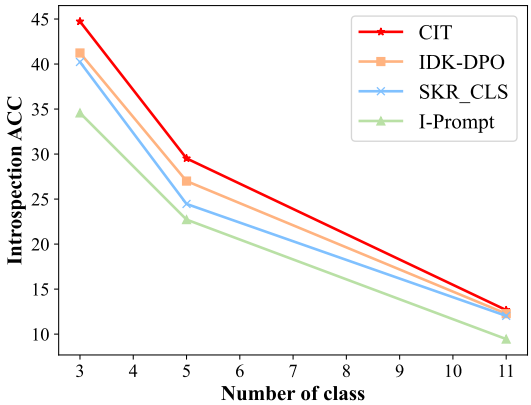


Figure 4: Results of CIT method and other baselines on TriviaQA data across three, five, and eleven classes.

Class Number of Confidence level We adjusted the division of confidence levels within the uncertain class to form new classification. When the entire uncertain class is treated as one class, the task becomes a three-class classification (*known*, *unknown*, *uncertain*). And if each CRes value (1∼9) in uncertain class is treated as one single class, the task becomes an eleven-class classification. We constructed the traning and test data similar to that in Section 2.3 and trained the same LLM using CIT method and baselines. The results are shown in Figure 4.

As shown in Figure 4, more classes (eleven) indicates a smaller gap between different confidence level, which increases the impact of sampling errors and makes it more challenging for learning of LLM. This is why we did not design CIT based on an eleven-class classification. Moreover, although the three-class ACC is higher, it lacks specific confidence information for the uncertain question, offering limited assistance to users. Furthermore, Our CIT method achieves the best results across experiments with various numbers of class, demonstrating its robustness.

Model Size and Type We introduce two additional LLMs, Llama2-7B-Chat and Llama2-13B-Chat (Touvron et al., 2023), and conduct experiments with three, five, and eleven confidence level classes, as shown in Figure 5.

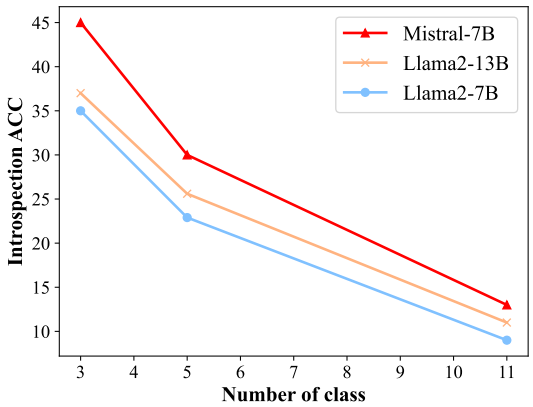
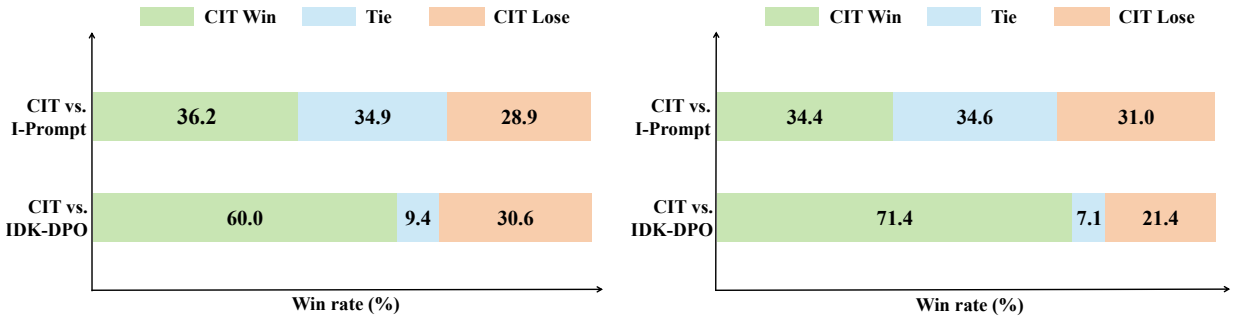


Figure 5: Results of of Mistral-7B, Llama2-7B, and Llama2-13B on TriviaQA after training with CIT cross three, five, and eleven classes.

We find that Llama2-13B-Chat outperforms Llama2-7B-Chat across all classification tasks, which indicates that larger LLMs exhibit stronger classification abilities after training with CIT. Different types of LLMs also show significant variation after CIT training. For instance, Mistral-7B performs better than Llama2-7B after CIT. This may be related to Mistral's inherently stronger semantic understanding and generation capabilities (Jiang et al., 2023).



(a) **Win rate on TriviaQA**

(b) **Win rate on Natural Questions**

Figure 6: "Win" indicates that elevators consider the response from CIT to be more helpful. "Lose" indicates that other method's responses are more helpful. "Tie" means that the responses from both methods convey similar meanings.

4.2 Human evaluation of Helpfulness

Helpfulness (Bai et al., 2022) stresses that LLMs should provide accurate, useful information related to the question. Even if unable to answer correctly, the LLM should still offer relevant details. To achieve this, as mentioned above, we introduce the *uncertain* class along with prefixes, serving two main goals: offering the LLM more response options to avoid defaulting to *I don't know* and providing users with more helpful information. We conducted human evaluation experiments to compare responses generated by CIT and baseline methods, aiming to validate that CIT effectively preserves the helpfulness of the vanilla model.

We conduct the human evaluation of our CIT method, IDK-DPO and I-Prompt method. We present fifty question and responses from two methods side by side, asking three researchers specialized in natural language processing as evaluators to determine which response is more helpful for the given question or if there is no significant difference between the two. Evaluators need to consider differences of responses in terms of QA accuracy, relevance to the question and the richness of the information provided. The order of the responses is randomized to avoid position preference. Figure 6 summarizes the comparison. Human evaluation instruction is shown in the Appendix A.4.

From the experimental results, compared to the IDK-DPO method, the CIT method demonstrates a stronger ability to provide helpful information (with the rate of CIT wins greater than losses). Moreover, CIT method even outperforms the vanilla model (I-Prompt), with the proportion of wins exceeding losses. This indicates that our method, building on enhancing the LLM's ability to distinguish confidence level of questions, ensures the LLM's ability to provide helpful suggestions. More case studies are presented in Table 8.

4.3 Confidence and Correct Steps Percentage

Confidence of question is derived from sampling accuracy. For multi-step reasoning in mathematical problems, we are concerned whether confidence can be reflected within the reasoning path. We extracted sixty *uncertain* questions from the original GSM8K test set, along with the CIT method's responses to them, ensuring that these questions were correctly classified. Each confidence level includes twenty questions. We assigned the question-response pairs to three volunteers. Each volunteer was required to examine the proportion of correct reasoning steps to the total steps in the reasoning path. Afterward, they exchanged data and repeated the process. Here, one "step" refers to one calculation process (Wei et al., 2023). Finally, we calculate the average percentage of correct reasoning steps of total steps for each confidence level. Detailed instructions for the human evaluation can be found in the Appendix A.5.

CRes	$1 \sim 3$	$4 \sim 6$	$7 \sim 9$
Pct.	0.29	0.58	0.84
EM	10.00	35.00	65.00

Table 3: Average percentage (Pct.) of correct reasoning steps of total steps and generation EM for three uncertain confidence level.

As shown in Table 3, there is a positive corre-

lation between the percentage of correct steps in the reasoning path and the confidence level. The higher the confidence level, the greater the percentage of correct steps in the reasoning path, which also increases the likelihood of a correct response. Consequently, the EM also improves as the confidence level increases.

4.4 Changes of Generation Exact Match

Methods	TQA-EM	NQ-EM
I-Prompt	57.48	52.95
CIT-Golden	44.23	22.41
CIT	55.64	49.64

Table 4: Generation Exact Match of different methods.

In above Table 2 in Section 3.4, we observe that the EM-K and EM-U scores of CIT are significantly higher than those of the vanilla model (I-Prompt). We aim to verify whether this improvement stems from an actual enhancement in QA ability or merely from the vanilla model classifying *unknown* questions into the *known* and *uncertain* class.

Therefore, we compared the QA accuracy (EM) across the entire test set of TQA and NQ, regardless of question class (including *unknown*). As shown in Table 4, the overall QA accuracy has unexpectedly decreased compared to the vanilla model. Then,we add a baseline, CIT-Golden, that trained by the golden answers of questions from our training set, where the accuracy drop is even more significant than CIT. This indicates that the improvement of EM-K and EM-U of CIT mainly results from accurately excluding *unknown* questions. On the other hand, the parameters of Mistral seem to be in a local optimum, where small-scale training is likely to reduce performance [4]. However, using self-generated data can mitigate this issue. CIT utilizes the LLM's own outputs for training, ensuring that the expected responses are more closely aligned with the expression style the LLM is accustomed to.

[4]We overfit the model on the test set using the same training method, achieving EM scores of 91.09 and 89.59, which demonstrates that our training algorithm has no flaws.

5 Related Work

5.1 Factual hallucinations in LLMs

Factual hallucination occurs that the LLM's response is inconsistent with reality. LLMs may confidently generate responses while being unaware of their own errors. (Huang et al., 2023; Ji et al., 2023; Cheng et al., 2023; Varshney et al., 2023; Manakul et al., 2023). This may be due to misinformation in pretraining data (Li et al., 2023a,b; Manakul et al., 2023), LLMs often contain a significant amount of erroneous knowledge. Moreover, most tuning methods for LLMs that align with human preferences compel the model to complete sentences, regardless of whether the LLM possesses the necessary knowledge. (Perez et al., 2022; Yin et al., 2023; Wei et al., 2024; Zhang et al., 2024).

5.2 Confidence calibration of LLMs

Confidence Calibration is an important way to mitigate factual hallucinations, ensuring that the confidence level assigned to a prediction accurately represents the probability of its correctness(Guo et al., 2017; Tian et al., 2023). Previous work can be categorized into external and internal calibration methods. External calibration methods rely on external models (like BERT (Devlin et al., 2019)) to estimate LLM's confidence (Mallen et al., 2023; Cao, 2023; Wang et al., 2023b), leaving LLMs with a lack of ability to make independent judgments. Internal methods teach the LLM to express confidence with statements like *I don't know* or *I am not sure* when the LLM lacks relevant knowledge(Lin et al., 2022; Cheng et al., 2024; Xu et al., 2024). However, these methods may lead the LLM to habitually refuse to answer most questions, failing to provide any further information. Our method ensures that the LLM can actively express its level of confidence while providing helpful responses as well.

6 Conclusions

We propose Confidence Introspection Training to enable the LLM to discern what question they know, what they don't know and what they are uncertain about. Experimental results demonstrate that our proposed method not only enhances the model's reliability by accurately expressing its confidence, but also maintains the helpfulness of the vanilla model.

Limitations

The limitations of our work can be summarized in two main aspects. First, we mainly focus on enabling the LLM to generate its own assessments of confidence. Whether this approach can be combined with classification models remains a research topic worth exploring. Second, our experiments were conducted mainly on models with size of 13B. Due to resource constraints, we did not perform experiments on larger models, and thus it requires to figure out whether our methods are generalized to larger-scale models.

References

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Kamile Lukosuite, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemi Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. 2022. Constitutional ai: Harmlessness from ai feedback. Preprint, arXiv:2212.08073.

Lang Cao. 2023. Learn to refuse: Making large language models more controllable and reliable through knowledge scope limitation and refusal mechanism. Preprint, arXiv:2311.01041.

Qinyuan Cheng, Tianxiang Sun, Xiangyang Liu, Wenwei Zhang, Zhangyue Yin, Shimin Li, Linyang Li, Zhengfu He, Kai Chen, and Xipeng Qiu. 2024. Can ai assistants know what they don't know? Preprint, arXiv:2401.13275.

Qinyuan Cheng, Tianxiang Sun, Wenwei Zhang, Siyin Wang, Xiangyang Liu, Mozhi Zhang, Junliang He, Mianqiu Huang, Zhangyue Yin, Kai Chen, and Xipeng Qiu. 2023. Evaluating hallucinations in chinese large language models. Preprint, arXiv:2310.03368.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On calibration of modern neural networks. In Proceedings of the 34th International Conference on Machine Learning, volume 70 of Proceedings of Machine Learning Research, pages 1321–1330. PMLR.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2023. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. Preprint, arXiv:2311.05232.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. ACM Computing Surveys, 55(12):1–38.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. Preprint, arXiv:2310.06825.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

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

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. Preprint, arXiv:2302.09664.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. Transactions of the Association for Computational Linguistics, 7:452–466.

Daliang Li, Ankit Singh Rawat, Manzil Zaheer, Xin Wang, Michal Lukasik, Andreas Veit, Felix Yu, and Sanjiv Kumar. 2023a. Large language models with controllable working memory. In Findings of the Association for Computational Linguistics: ACL 2023, pages 1774–1793, Toronto, Canada. Association for Computational Linguistics.

Sha Li, Chi Han, Pengfei Yu, Carl Edwards, Manling Li, Xingyao Wang, Yi Fung, Charles Yu, Joel Tetreault, Eduard Hovy, and Heng Ji. 2023b. Defining a new NLP playground. In Findings of the Association for Computational Linguistics: EMNLP 2023, pages 11932–11951, Singapore. Association for Computational Linguistics.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Teaching models to express their uncertainty in words. Transactions on Machine Learning Research.

Hao Liu, Carmelo Sferrazza, and Pieter Abbeel. 2023. Chain of hindsight aligns language models with feedback. arXiv preprint arXiv:2302.02676.

Matéo Mahaut, Laura Aina, Paula Czarnowska, Momchil Hardalov, Thomas Müller, and Lluís Màrquez. 2024. Factual confidence of llms: on reliability and robustness of current estimators. Preprint, arXiv:2406.13415.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 9802–9822, Toronto, Canada. Association for Computational Linguistics.

Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. 2023. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. Preprint, arXiv:2303.08896.

Gerald E. Myers. 1986. Introspection and self-knowledge. American Philosophical Quarterly, 23(2):199–207.

Ethan Perez, Sam Ringer, Kamilė Lukošiūtė, Karina Nguyen, Edwin Chen, Scott Heiner, Craig Pettit, Catherine Olsson, Sandipan Kundu, Saurav Kadavath, Andy Jones, Anna Chen, Ben Mann, Brian Israel, Bryan Seethor, Cameron McKinnon, Christopher Olah, Da Yan, Daniela Amodei, Dario Amodei, Dawn Drain, Dustin Li, Eli Tran-Johnson, Guro Khundadze, Jackson Kernion, James Landis, Jamie Kerr, Jared Mueller, Jeeyoon Hyun, Joshua Landau, Kamal Ndousse, Landon Goldberg, Liane Lovitt, Martin Lucas, Michael Sellitto, Miranda Zhang, Neerav Kingsland, Nelson Elhage, Nicholas Joseph, Noemí Mercado, Nova DasSarma, Oliver Rausch, Robin Larson, Sam McCandlish, Scott Johnston, Shauna Kravec, Sheer El Showk, Tamera Lanham, Timothy Telleen-Lawton, Tom Brown, Tom Henighan, Tristan Hume, Yuntao Bai, Zac Hatfield-Dodds, Jack Clark, Samuel R. Bowman, Amanda Askell, Roger Grosse, Danny Hernandez, Deep Ganguli, Evan Hubinger, Nicholas Schiefer, and Jared Kaplan. 2022. Discovering language model behaviors with model-written evaluations. Preprint, arXiv:2212.09251.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. Preprint, arXiv:2305.18290.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D. Manning. 2023. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. Preprint, arXiv:2305.14975.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. Preprint, arXiv:2307.09288.

Neeraj Varshney, Wenlin Yao, Hongming Zhang, Jianshu Chen, and Dong Yu. 2023. A stitch in time saves nine: Detecting and mitigating hallucinations of llms by validating low-confidence generation. Preprint, arXiv:2307.03987.

Cunxiang Wang, Sirui Cheng, Qipeng Guo, Yuanhao Yue, Bowen Ding, Zhikun Xu, Yidong Wang, Xiangkun Hu, Zheng Zhang, and Yue Zhang. 2023a. Evaluating open-QA evaluation. In Thirty-seventh

Conference on Neural Information Processing Systems Datasets and Benchmarks Track.

Yile Wang, Peng Li, Maosong Sun, and Yang Liu. 2023b. Self-knowledge guided retrieval augmentation for large language models. In Findings of the Association for Computational Linguistics: EMNLP 2023, pages 10303–10315, Singapore. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models. Preprint, arXiv:2201.11903.

Jerry Wei, Da Huang, Yifeng Lu, Denny Zhou, and Quoc V. Le. 2024. Simple synthetic data reduces sycophancy in large language models. Preprint, arXiv:2308.03958.

Hongshen Xu, Zichen Zhu, Situo Zhang, Da Ma, Shuai Fan, Lu Chen, and Kai Yu. 2024. Rejection improves reliability: Training llms to refuse unknown questions using rl from knowledge feedback. Preprint, arXiv:2403.18349.

Zhangyue Yin, Qiushi Sun, Qipeng Guo, Jiawen Wu, Xipeng Qiu, and Xuanjing Huang. 2023. Do large language models know what they don't know? In Findings of the Association for Computational Linguistics: ACL 2023, pages 8653–8665, Toronto, Canada. Association for Computational Linguistics.

Hanning Zhang, Shizhe Diao, Yong Lin, Yi Fung, Qing Lian, Xingyao Wang, Yangyi Chen, Heng Ji, and Tong Zhang. 2024. 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 7113–7139, Mexico City, Mexico. Association for Computational Linguistics.

A Prompts and prefixes

A.1 Prefixes

Uncertain Prefixes :

- I'm not sure of the answer to that, but here's what I would suggest these.
- I can't give you a definitive answer, but I can offer some insights based on my knowledge.
- I'm not sure how to answer that, but here's what I think.

Unknown Prefixes:

- I don't know the answer to that, but I would suggest these.
- I don't have personal experience with that, but here are some general tips.
- I'm not knowledgeable about that topic, but I can try to find the answer for you.

A.2 Instructions of I-Prompt and other methods

The prompt we used for testing is: `Please answer the following question, your answer should be as simple as possible. If you possess relevant knowledge, respond with confidence. If you are unfamiliar with the required knowledge, please honestly inform the user.Please answer this question:{question}`.

A.3 Evaluation of Introspection Prompting

In addition to the prefixes designed in our experiment to indicate *unknown*, if the following statements appear, they can also be considered as the model expressing of *unknown* or *uncertain*. These statements are summarized from the outputs of the vanilla model , as shown in Table 5.

CRes	Statements
0	I am not an expert in the specific area.
1∼3	My Confidence: Low.
4∼6	I'm not entirely confident about this answer.
7∼9	I am 90% confident that.

Table 5: Statements representing the confidence level of the vanilla model.

A.4 Instruction of Helpfulness Evaluation

Prompt Please evaluate the quality of these model responses based on the following criteria and clearly identify which model's response is better in terms of helpfulness.Helpfulness: When the model responds correctly, it should provide detailed information. When the responses are incorrect or when expressing uncertainty, the model should offer useful guesses or suggestions. Question: {question}.Response A:{res_A}.esponse B:{res_B}. After evaluating responses A and B, select one of the following options: (1) the output of the first method is more helpful than the second; (2) the output of the second method is more helpful than the first; (3) there is no significant difference between the two outputs.

Dataset	0	1	2	3	4	5	6	7	8	9	10
TQA	38.1	4.7	3.0	2.9	3.3	8.1	6.5	6.3	6.7	9.2	15.2
NQ	24.6	7.5	5.4	4.1	3.5	5.8	4.7	3.9	5.5	6.3	14.8
GSM8K	26.1	16.5	11.5	9.9	8.6	8.2	5.4	4.2	4.5	2.9	2.4

Table 6: The proportion of uncertain questions with CRes ranging from 0.1 to 0.9 in the test sets of three datasets: TriviaQA (TQA), Natural Questions (NQ), and GSM8K.

A.5 Instruction of Correct steps Evaluation

Instruction The following is a math problem and its response. Carefully review each calculation step in the response for logical coherence and accuracy, determining whether each step is reasonable and correct. If an error in a previous step causes an error in the current calculation, the current step should also be marked incorrect. Finally, provide the percentage of correct calculation steps to the total number of steps in this response.

B Confidence distribution of Data

Table 6 shows the data distribution across various CRes values.

C Details of CIs Datasets and Training

C.1 Statistics in the dataset

Split	Confidence	TQA	NQ	GSM8K
Test	Known	1,000	500	31
	Uncertain	3,000	1,500	450
	Unknown	1,000	500	150
	Total	**5,000**	**2,500**	**631**

Split	Classification	TQA
Train	Known	10,000
	Uncertain	30,000
	Unknown	10,000
	Total	**50,000**
Dev	Known	300
	Uncertain	900
	Unknown	300
	Total	**1,500**

Table 7: Details of the train, dev and test set of CIs dataset. The numbers listed in the table represent the number of data.

C.2 Details of Training

The β value for DPO is set 0.1, with 1 training epochs after SFT. During testing, we use the vllm[5] framework and employ greedy decoding for generation. All methods are evaluated on test set in our KIs dataset.

In actual training, we use sub-prefixes such as "My confidence is 0.2", "My confidence is 0.5" and "My confidence is 0.8" to represent the three confidence levels of the uncertain class (using the median of each CRes category divided by 10). Training in this way enables most methods to achieve better classification performance. After the model generates responses, we replace the sub-prefixes with those in Table 1 to help user better understand the model's confidence level. Sub-prefixes are simply symbolic of confidence levels. The final replacement does not affect model performance and makes the output more user-friendly.

D Case studies of CIT and baselines

As shown in Table 8, the LLM trained with our Confidence Introspection Training (CIT) method is capable of providing information relevant to the question. Even with a confidence level of only 0.5 (for Question1), the responses can still include the correct answer. Compared to I-Prompt, our method maintains the fundamental function of the LLM to provide useful information to users and presents the model's confidence level.

Compared to IDK-DPO, it is evident that the CIT method provides substantially more information when facing Questions 2 and 3. Although the LLM does not know the golden answer for Question 3, the LLM trained with CIT still offers substantial helpful information, whereas IDK-DPO merely responds with "I don't know." This demonstrates that the CIT method can effectively prevent the model from becoming overly conservative, thereby balancing reliability and helpfulness.

[5] https://github.com/vllm-project/vllm

Question1	Whose backing band is known as The Miami Sound Machine?	
Methods	**Responses**	**EM**
I-Prompt	The Miami Sound Machine is the backing band for the singer Gloria Estefan. They gained international fame in the late 1980s and early 1990s with their fusion of Latin rhythms and pop music. I am confident in my answer.	✓
IDK-DPO	I'm not sure how to answer that.	✗
CIT	I'm not sure of the answer to that, but here's what I would suggest these.My confidence is 0.5.The Miami Sound Machine is a band that was formed in Miami, Florida, in the late 1960s. They gained international fame in the 1980s with their fusion of Latin rhythms, jazz, and pop music. The band's most famous member is probably lead singer Gloria Estefan. Other notable members include her husband, Emilio Estefan, who was the band's founder and producer, and drummer Enrique Rodriguez. The Miami Sound Machine disbanded in the late 1990s but reunited in 2006 for a world tour.	✓
Question2	Michelangelo Merisi (or Amerighi) is the birth name of which artist?	
Methods	**Responses**	**EM**
I-Prompt	Caravaggio, Confidence: Low. The name Michelangelo Merisi (or Amerighi) is not associated with Caravaggio. Michelangelo Merisi da Caravaggio was an Italian painter active in the late 16th and early 17th centuries, best known for his dramatic use of chiaroscuro, a technique that employs strong contrasts between light and dark. Caravaggio's full name was actually Michelangelo Merisi da Caravaggio. However, I cannot confirm that the given name is the birth name of Caravaggio with absolute certainty. It is commonly accepted that this is his full name, but there may be some variation in the sources. I would recommend double-checking with a reliable art history reference for the most accurate information.	✓
IDK-DPO	Michelangelo Merisi is the birth name of the artist Caravaggio.	✓
CIT	Michelangelo Merisi, also known as Il Caravaggio, was an Italian painter. He is famous for his dramatic use of chiaroscuro, a technique that emphasizes the contrast between light and dark in a painting. Caravaggio's works often depict scenes of religious and mythological subjects, as well as everyday life in 16th-century Italy. Some of his most famous paintings include "The Calling of St. Matthew", "The Supper at Emmaus" and "The Entombment of Christ".	✓
Question3	What is the significance of 16th June 1904?	
Methods	**Responses**	**EM**
I-Prompt	I am not confident in answering this question without further context. The significance of June 16, 1904, depends on the specific historical or cultural context in which it is being considered. It could be a significant date in the history of a particular country, organization, or individual. For example, it could be the date of a battle, a treaty signing, a birth or death, or an important event in the history of a particular field or industry. Without more information, it is not possible to determine the significance of this date with certainty.	✗
IDK-DPO	I'm not knowledgeable about that topic.	✗
CIT	I don't know the answer to that, but I would suggest these. The significance of June 16th, 1904, is that it marks the beginning of the Mexican Revolution. On this day, the Mexican Liberal Party, led by Francisco Madero, nominated him as their presidential candidate. Madero's election in November 1912 marked the end of the long dictatorship of Porfirio Diaz. However, Madero's presidency was short-lived as he was assassinated in February 1913. The Mexican Revolution, which lasted from 1910 to 1920, brought about significant political and social changes in Mexico.	✗

Table 8: The responses of Mistral-7B after our Confidence Introspection Training (CIT), IDK-DPO(Cheng et al., 2024) and I-Prompt. ✓ and ✗ denote whether the response contains the golden answer.