

Think Twice Before Trusting: Self-Detection for Large Language Models through Comprehensive Answer Reflection

Moxin Li¹, Wenjie Wang^{1*}, Fuli Feng^{2,3*}, Fengbin Zhu¹, Qifan Wang⁴
Tat-Seng Chua¹

¹National University of Singapore, ²University of Science and Technology of China

³Institute of Dataspace, Hefei, Anhui, China, ⁴Meta AI

limoxin@u.nus.edu , wangwenjie@u.nus.edu, fulifeng93@gmail.com,
fengbin@nus.edu.sg, wqfcr@meta.com, dcscts@nus.edu.sg

Abstract

Self-detection for Large Language Models (LLMs) seeks to evaluate the trustworthiness of the LLM’s output by leveraging its own capabilities, thereby alleviating the issue of output hallucination. However, existing self-detection approaches only retrospectively evaluate answers generated by LLM, typically leading to the over-trust in incorrectly generated answers. To tackle this limitation, we propose a novel self-detection paradigm that considers the comprehensive answer space beyond LLM-generated answers. It thoroughly compares the trustworthiness of multiple candidate answers to mitigate the over-trust in LLM-generated incorrect answers. Building upon this paradigm, we introduce a two-step framework, which firstly instructs LLM to reflect and provide justifications for each candidate answer, and then aggregates the justifications for comprehensive target answer evaluation. This framework can be seamlessly integrated with existing approaches for superior self-detection. Extensive experiments on six datasets spanning three tasks demonstrate the effectiveness of the proposed framework.

1 Introduction

Large Language Model (LLM) typically suffers from the hallucination issue, (Zhang et al., 2023c; Li et al., 2023a; Golovneva et al., 2022; Bang et al., 2023), which significantly undermines the trustworthiness of LLM’s outputs. A promising research direction for evaluating the output trustworthiness and identifying incorrect outputs is self-detection (Zhao et al., 2023c; Miao et al., 2023; Manakul et al., 2023). Given a question, self-detection aims to leverage LLM’s own ability to evaluate the trustworthiness of its generated answers, without relying on external knowledge sources or specifically trained detection models. This paper investigates self-detection methods tailored for black-box API

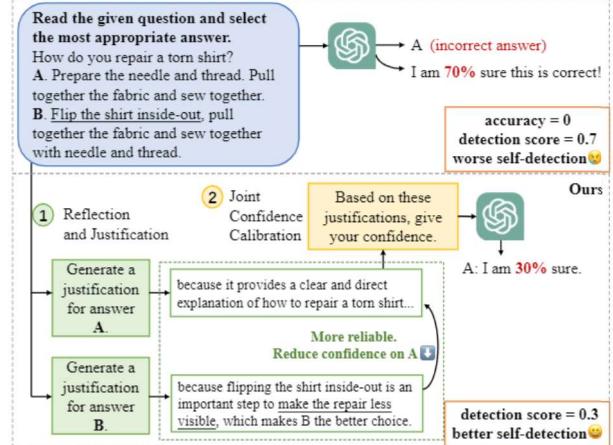


Figure 1: An illustration of *Think Twice before Trusting* framework for mitigating the over-trust issue in LLM self-detection. LLM is instructed to reflect and generate justifications on the trustworthiness of each answers before evaluating the trustworthiness of the target answer.

LLMs due to their excellent performance and the inherent challenge posed by limited output information (Achiam et al., 2023; OpenAI, 2024).

Previous studies in self-detection can be broadly categorized into two paradigms (*cf.* Figure 2). The first paradigm is confidence calibration, aiming to estimate LLM’s confidence on the generated answer to align with the actual answer accuracy via multi-answer sampling and aggregation (Xiong et al., 2023; Tian et al., 2023b; Si et al., 2022; Jiang et al., 2023). The second one is self-evaluation, which directly examines the compatibility of question and answer by designing various prompt strategies (Miao et al., 2023; Kadavath et al., 2022; Weng et al., 2023). These two paradigms have also been combined to enhance self-detection capabilities (Chen and Mueller, 2023; Ren et al., 2023a).

However, both self-detection paradigms have shown a significant drawback: an inclination towards over-trusting the incorrect answers generated by LLM (Si et al., 2022; Xiong et al., 2023; Jiang et al., 2023; Kadavath et al., 2022). We argue that one reason may be that both paradigms merely eval-

*Corresponding author.

在信任之前三思：通过全面答案反思进行大型语言模型的自我检测

李莫辛，王文杰，冯福丽，朱峰斌，王奇凡，蔡达森

¹新加坡国立大学，中国科学技术大学数据空间研究院，安徽省合肥，中国，Meta AI

limoxin@u.nus.edu, wangwenjie@u.nus.edu, fulifeng93@gmail.com,
fengbin@nus.edu.sg, wqfc@meta.com, dcscts@nus.edu.sg

limoxin@u.nus.edu, 王文杰@u.nus.edu, fulifeng93@gmail.com, 风斌

@nus.edu.sg, wqfc@meta.com, dcscts@nus.edu.sg

摘要 自我检测大型语言模型 (LLMs) 旨在通过利用自身的功能来评估LLM输出的可信度，从而减轻输出幻觉的问题。然而，现有的自我检测方法通常只对LLM生成的答案进行回顾性评估，这往往导致对错误生成的答案过度信任。为了解决这一局限性，我们提出了一种新颖的自我检测范式，该范式考虑了超出LLM生成答案的全面答案空间。它详细比较多个候选答案的可信度，以减轻对LLM生成错误答案的过度信任。在此基础上，我们引入了一个两步框架，首先指导LLM对每个候选答案进行反思并提供理由，然后汇总这些理由以进行全面的目标答案评估。该框架可以无缝集成到现有的自我检测方法中。在涵盖三个任务的六个数据集上进行的广泛实验证明了所提出框架的有效性。

1 引言

大型语言模型 (LLM) 通常存在幻觉问题 (张等, 2023c; 李等, 2023a; 戈洛夫涅娃等, 2022; 邦等, 2023)，这严重削弱了LLM输出的可信度。评估输出可信度和识别错误输出的一个有前景的研究方向是自我检测 (赵等, 2023c; 苗等, 2023; 马纳库尔等, 2023)。给定一个问题，自我检测旨在利用LLM自身评估其生成答案可信度的能力，而不依赖于外部知识源或专门训练的检测模型。本文研究了针对黑盒 API 的自检测方法。

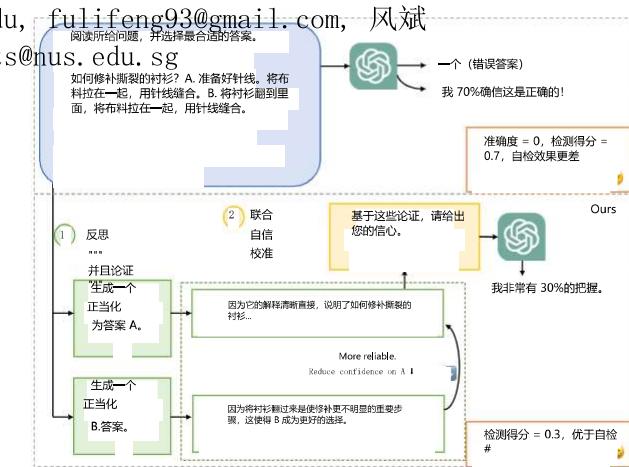


图 1：展示“三思而后信”框架，用于缓解在LLM自我检测中过度信任的问题。LLM被指导在评估目标答案的可信度之前，反思并生成对每个答案可信度的论证。

由于它们卓越的表现以及有限的输出信息带来的固有挑战 (Achiam 等人, 2023; OpenAI, 2024)。

之前在自我检测领域的研究可以大致分为两种范式（参见图 2）。第一种范式是置信度校准，旨在通过多答案采样和聚合来估计LLM对生成的答案的置信度，以使其与实际答案的准确性相匹配 (Xiong 等, 2023; Tian 等, 2023b; Si 等, 2022; Jiang 等, 2023)。第二种范式是自我评估，通过设计各种提示策略直接检验问题和答案的兼容性 (Miao 等, 2023; Kadavath 等, 2022; Weng 等, 2023)。这两种范式也已被结合以增强自我检测能力 (Chen 和 Mueller, 2023; Ren 等, 2023a)。

然而，这两种自我检测范式都显示出显著的缺点：倾向于过度信任由LLM (Si 等, 2022; Xiong 等, 2023; Jiang 等, 2023; Kadavath 等, 2022) 生成的错误答案。我们认为，其中一个原因可能是因为这两种范式仅仅评估——

* 对应作者。

uate LLM-generated answers, while LLM contains an inherent bias towards trusting its own generations (Mielke et al., 2022; Lin et al., 2022a), leading to serious over-trust in LLM-generated incorrect answers. An ideal self-detection paradigm should consider a more comprehensive answer space beyond LLM’s generations. By evaluating on other potentially correct answers in a broader answer space, the strong validity in these answers can counterbalance the excessive trust in the incorrect LLM answers, thus alleviating the over-trust issue.

In this light, we introduce a new comprehensive answer evaluation paradigm involving the consideration of multiple candidate answers in the answer space to enhance self-detection (*cf.* Figure 2). This paradigm meticulously evaluates each answer’s trustworthiness as a correct answer to the question and aggregates these evaluations to enhance the self-detection of the target LLM answer. The biased trust in the LLM-generated incorrect answers can be alleviated through the trustworthiness comparison with other more trustable answers. Our preliminary experiments reveal the efficacy of considering more comprehensive answers to confront over-trust (*cf.* Section 2). To summarize, two key considerations arise to instantiate this new paradigm: 1) resisting the inherent bias of LLM to precisely evaluate the trustworthiness of each question-answer pair, and 2) aggregating these evaluations in the trustworthiness evaluation of the target answer.

To this end, we present a novel self-detection framework to tackle the over-trust issue of LLMs, named Think Twice before Trusting (T^3) (*cf.* Figure 1). Our framework pushes LLM to reflect and justify from different answers’ perspectives before arriving at the trustworthiness on the target answer. Firstly, the LLM is instructed to generate justifications regarding the potential correctness of each answer. Subsequently, a prompt-based method is employed to integrate these justifications into joint evaluation for the target answer. Extensive experiments on six datasets across three tasks on three different LLMs show improved performance of T^3 over methods from existing paradigms. Notably, T^3 can be combined with existing methods for superior self-detection. Our analysis also reveals T^3 ’s strong robustness and effective over-trust mitigation. Our contributions are three-fold.

- We introduce a novel self-detection paradigm for mitigating the over-trust issue in LLM, ad-

dressing the limitation of existing paradigms by reflection in the broader answer space.

- We present a novel T^3 framework to implement the comprehensive answer evaluation paradigm, which can be seamlessly integrated with existing self-detection methods.
- We conduct extensive experiments on three NLP tasks with six datasets, validating the rationality and effectiveness of the proposed framework.

2 Problem Formulation

LLM Self-Detection. We formulate the task of self-detection for LLM as follows. Given the input comprising of question q combined with prompt p , which consists of an instruction and optional in-context examples, LLM can generate the answer a (Brown et al., 2020), denoted as the target answer. Thereafter, self-detection aims to evaluate the trustworthiness of a by LLM’s own ability, generally in the form of a detection score $c \in \mathcal{R}$ ¹. The detection score c can be used for indicating the actual accuracy of a , where low c values indicate potentially incorrect answers. Denoting the self-detection strategy as a function $SD(\cdot)$, this process can be abstracted as

$$a = LLM(p(q)), \quad (1)$$

$$c = SD(LLM(\cdot), q, a). \quad (2)$$

In the following, we illustrate the existing two paradigms for self-detection, *i.e.*, confidence calibration and self-evaluation, and introduce our proposed comprehensive answer evaluation paradigm.

Confidence Calibration. Confidence calibration aims to estimate LLM’s level of certainty on the answer a , *e.g.*, estimating the LLM output probability of a , where the obtained confidence score as the detection score c aims to calibrate with the actual answer accuracy. Xiong et al. conclude a general three-step confidence calibration process for LLM as *prompting*, *answer sampling*, *aggregation* (*cf.* Figure 2). Denoting the prompt for confidence calibration as $p^c(\cdot)$ and the aggregation function as $Aggr(\cdot)$, this paradigm can be abstracted as,

$$c = Aggr(a, \{a_1, \dots, a_D\}), \quad (3)$$

$$\text{where } a_i = LLM(p^c(q)), i \in \{1, \dots, D\}.$$

¹If the result of self-detection is a class label (trustable or untrustable), the detection score can be formulated as 1 or 0.

2022) 对 1001#生成的答案进行评估，而 1002# 则存在倾向于信任自身生成的答案的固有偏差 (Mielke 等人, 2022; Lin 等人, 2022a)，导致对 1003#生成的错误答案过度信任。理想的自我检测范式应考虑超出 1004#生成答案的更广泛的答案空间。通过在更广泛的答案空间中评估其他可能的正确答案，这些答案的强有效性可以抵消对错误 1005#答案的过度信任，从而缓解过度信任问题。

在此背景下，我们提出了一种新的全面答案评估范式，该范式涉及在答案空间中考虑多个候选答案，以增强自我检测（参见图 2）。该范式细致地评估每个答案作为正确答案的可靠性，并将这些评估汇总以增强对目标答案 LLM 的自我检测。通过与其他更可靠的答案的可靠性比较，可以缓解对 LLM 生成的错误答案的偏见。我们的初步实验揭示了考虑更全面答案以应对过度信任的有效性（参见图 2）。总之，有两个关键考虑因素可以实施这一新范式：1) 抵制 LLM 对每个问题-答案对可靠性进行精确评估的固有偏见，2) 将这些评估汇总到目标答案的可靠性评估中。

为此，我们提出了一种新颖的自我检测框架来解决 LLMs 的过度信任问题，命名为“三思而后信” (T)（参见图 1）。我们的框架促使 LLM 从不同答案的角度进行反思和论证，然后再对目标答案的可信度进行判断。首先，指导 LLM 就每个答案可能正确性生成论证。随后，采用基于提示的方法将这些论证整合到对目标答案的联合评估中。在三个不同 LLMs 的六个数据集上进行的广泛实验表明，T 在三个任务上的性能优于现有范式的方法。值得注意的是，T 可以与现有方法结合以实现更优的自我检测。我们的分析还揭示了 T 强大的鲁棒性和有效的过度信任缓解能力。我们的贡献有三点。

通过反思在更广泛的答案空间中克服现有范式的局限性。

- 我们提出了一种新的 T 框架，用于实现全面的答案评估范式，该范式可以无缝集成到现有的自检测方法中。
- 我们对三个自然语言处理任务进行了广泛的实际实验，使用六个数据集验证了所提出框架的合理性和有效性。

2 问题定义

LLM 自我检测。我们将 LLM 的自我检测任务表述如下。给定输入包含问题 q 和提示 p ，其中提示 p 由指令和可选的上下文示例组成，LLM 可以生成答案 a (Brown 等人, 2020 年)，记为目标答案。之后，自我检测旨在通过 LLM 自身的检测能力来评估答案 a 的可信度，通常以检测分数 $c \in \mathbb{R}$ 的形式表示。检测分数 c 可以用来指示答案 a 的实际准确性，其中低 c 值表示答案可能不正确。将自我检测策略表示为函数 $SD(\cdot)$ ，此过程可以抽象为：

```
```plaintext
a = LLM (p(q)), (1) c = SD(LLM (\cdot), q,
a). (2)
```

```

以下，我们将阐述现有的两种自我检测范式，即置信度校准和自我评估，并介绍我们提出的综合答案评估范式。

```
```plaintext
a = LLM (p(q)) , (1) c = SD(LLM
(\cdot) , q, a) . (2)
```

```

自信校准。自信校准

旨在估计 LLM 对答案 a 的置信度，例如，估计 a 的 LLM 输出概率，其中所获得的置信度得分作为检测得分 c ，旨在与实际答案的准确性进行校准。Xiong 等人总结了针对 LLM 的一般三步置信度校准过程，即提示、答案采样、聚合（参见图 2）。将置信度校准的提示表示为 $p(\cdot)$ ，聚合函数表示为 $Aggr(\cdot)$ ，这种范式可以抽象为，

```
c = Aggr(a, {a, ..., a}), (3) 其中 a=
LLM (p(q)), i ∈ {1, ..., D}.
```

如果自我检测的结果是类别标签（可信或不可信），则检测分数可以表示为 1 或 0。

- 我们提出了一种新颖的自我检测范式，用于缓解 LLM 中的过度信任问题

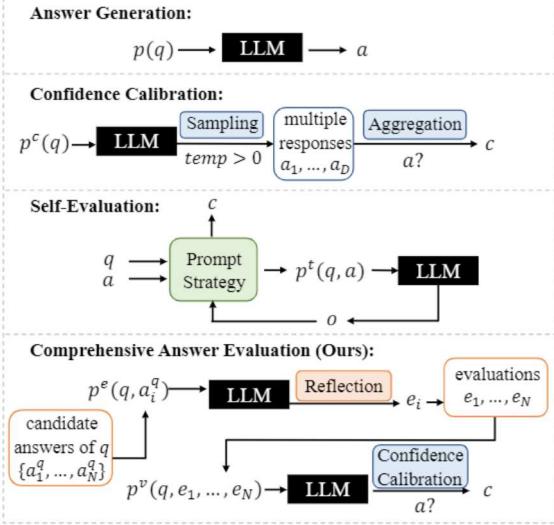


Figure 2: Two existing paradigms of self-detection and our new comprehensive answer evaluation paradigm.

where $D > 1$ refers to the number of sampled answers. For example, self-consistency (Wang et al., 2022; Si et al., 2022) aggregates the probability of a in the sampled outputs of $p(q)$ (e.g., using nucleus sampling (Holtzman et al., 2020)). Formally,

$$c = \frac{\sum_{i=1}^D \mathbb{1}(a_i = a)}{D}, \quad (4)$$

where $a_i = \text{LLM}(p(q))$, $i \in \{1, \dots, D\}$.

Besides, the Top- K verbalized methods (Lin et al., 2022a; Tian et al., 2023b) leverage a well-designed prompt p^b (cf. Appendix A.3) to instruct the LLM to sample the K most likely answers and output their corresponding probabilities in one response:

$$[\{a_1, c_1\}, \dots, \{a_K, c_K\}] = \text{LLM}(p^b(q)). \quad (5)$$

where $[\cdot]$ denotes the concatenation of the K most likely answers with their probabilities. The probability of a in the response is utilized as its detection score c ($c = 0$ if a is not in the K answers).

However, confidence calibration methods are observed with severe over-trust issue on LLM, assigning high confidence score in some incorrectly generated answers (Si et al., 2022; Xiong et al., 2023). In fact, LLM has a bias to blindly trust its generated answers, leading to difficulties in distinguishing the correctness of its generated answers (Huang et al., 2023b; Ling et al., 2023; Mielke et al., 2022; Ren et al., 2023b). Although some attempts have been made to reduce high confidence in LLM and achieve better calibration (Jiang et al., 2023; Zhao et al., 2024), the over-trust issue still remains a severe problem towards effective self-detection.

Self-Evaluation. Self-evaluation methods concatenate q and a and leverage various designed prompts to instruct LLM in self-evaluating the correctness of a from different perspectives. The prompt strategy examines the matching of q, a by integrating the self-evaluation output o . Denoting the prompt strategy as a function $ES(\cdot)$, this process can be summarized as

$$c = ES(q, a, o), \quad (6)$$

where $o = \text{LLM}(p^t(q, a))$.

where p^t represents one prompt for self-evaluation.

The shortcoming of self-evaluation is that many approaches under this paradigm are specifically designed for the mathematical question answering task, including step-wise checking on Chain-of-Thoughts (CoT) reasoning (Miao et al., 2023), completing masked q using a (Weng et al., 2023), and natural program (Ling et al., 2023), limiting its applicability. The general method P(True) (Kadavath et al., 2022) is straightforward and still demonstrates over-trust to incorrect LLM-generated answers. It directly asks LLM whether a is the true answer to q via the prompt p^r (cf. Appendix A.3), and uses the probability of “True” in the sampled responses as c . Formally,

$$c = \frac{\sum_{i=1}^D \mathbb{1}(o_i = \text{True})}{D}, \quad (7)$$

where $o_i = \text{LLM}(p^r(q, a))$, $i \in \{1, \dots, D\}$.

The two paradigms can be combined for better self-detection (Xiong et al., 2023; Chen and Mueller, 2023; Ren et al., 2023a; Agrawal et al., 2023).

A New Comprehensive Answer Evaluation Paradigm. A notable limitation of the existing two paradigms is that their evaluation merely involves LLM-generated answers a_i , in which LLM may exhibit over-trust. We argue that such biased over-trust could be alleviated if LLM had thoroughly compared the trustworthiness of more candidate answers of q beyond LLM-generated answers. We consider the multi-choice question answering setting where a comprehensive answer space is provided.² If other answers in q 's answer space had a strong tendency to be correct, the high detection score for LLM-generated incorrect a could be diminished, reducing the over-trust issue.

²For other settings, the answer space can be obtained via answer retrieval or additional model prediction.

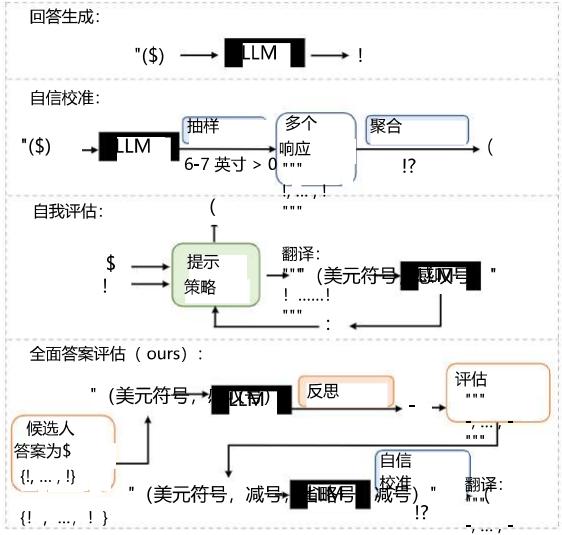


图2：两种现有的自检范式以及我们提出的新综合答案评估范式。

“其中， $D > 1$ 表示采样答案的数量。例如，自洽性（王等，2022；齐等，2022）聚合了 $p(q)$ 的采样输出中 a 的概率（例如，使用核采样（霍尔特兹曼等，2020））。形式上，”

$$c = \frac{P}{\sum_{i=1}^D p(a_i)} \quad (4)$$

其中 $a = \text{LLM}(p(q))$, $i \in \{1, \dots, D\}$ 。
此外，“Top-K 的表达方法（参数等 a）2022a；Tian 等人，2023b）利用精心设计的提示词 p （参见附录 A.3）来指导LLM采样 K 个最可能的答案，并在一个响应中输出它们对应的概率：

$$\dots [\{a, c\}, \dots \{a, c\}] = \text{LLM}(p(q)) \quad (5)$$

其中 $[\cdot]$ 表示将 K 个最可能的答案及其概率进行连接。[响应中] a 的概率被用作其检测分数 c （如果 a 不在 K 个答案中，则 $c=0$ ）。”

然而，在LLM上观察到的置信度校准方法存在严重的过度信任问题，对一些错误生成的答案给出了过高的置信度评分（Si 等，2022；Xiong 等，2023）。事实上，LLM倾向于盲目信任其生成的答案，导致难以区分其生成的答案的正确性（Huang 等，2023b；Ling 等，2023；Mielke 等，2022；Ren 等，2023b）。尽管已经尝试了一些方法来降低LLM的高置信度并实现更好的校准（Jiang 等，2023；Zhao 等，2024），但过度信任问题仍然是对有效自我检测的一个严重问题。

自我评估。自我评估方法将问题和答案结合，利用各种设计的提示来指导LLM从不同角度对答案的正确性进行自我评估。提示策略通过整合自我评估输出 o 来考察问题和答案的匹配度。将提示策略表示为函数 $ES(\cdot)$ ，这个过程可以概括为：

$$c = ES(q, a, o), \quad (6) \text{ 其中 } o = \text{LLM}(p(q, a)).$$

“其中 o represents 一个自我评估的提示。”
自我评估的不足之处在于，许多属于这一范式的方案专门针对数学问答任务设计，包括对思维链（CoT）推理的逐步检查（Miao 等，2023 年）、使用（Weng 等，2023 年）完成掩码 q ，以及自然程序（Ling 等，2023 年），这限制了其适用性。通用方法 $P(\text{True})$ （Kadavath 等，2022 年）简单直接，但仍然对由LLM生成的错误答案过于信任。它直接询问LLM a 是否是 q 的正确答案，通过提示 p （参见附录 A.3），并使用采样响应中 “True”的概率作为 c 。形式上，

$$c = \frac{P}{\sum_{i=1}^D \text{开启}=\text{True}} \quad (7)$$

$o = \text{LLM}(p(q, a))$ ，其中 $i \in \{1, \dots, D\}$ 。
两种范式可以结合使用，以实现更好的自我检测（Xiong 等，2023；Chen 和 Mueller，2023；Ren 等，2023a；Agrawal 等，2023）。

一种新的全面答案评估范式。现有两种范式的一个显著局限性是，它们的评估仅仅涉及由LLM生成的答案 a ，其中LLM可能表现出过度信任。我们认为，如果LLM能够彻底比较 q 中更多候选答案的可信度，超出LLM生成的答案，那么这种偏见的过度信任问题可以得到缓解。我们考虑了多选题答案设置，其中提供了一个全面的答案空间。如果 q 的答案空间中的其他答案有很强的正确倾向，那么对LLM生成的错误答案 a 的高检测分数可能会降低，从而减少过度信任问题。

对于其他设置，答案空间可以通过答案检索或额外模型预测获得。

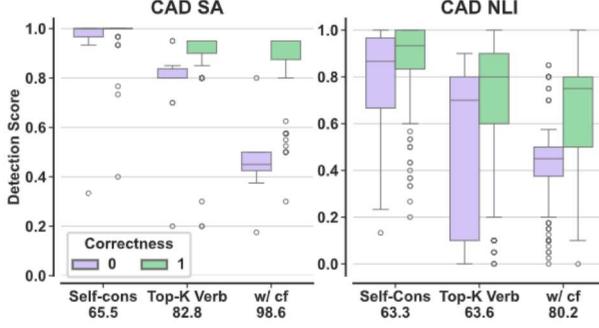


Figure 3: Comparison of self-detection methods on CAD. *w/ cf* denotes our strategy with counterfactual data. The AUROC values are shown in the x-axis. The boxes on the left and right represent the detection scores of incorrect and correct instances, respectively.

In this light, we propose a novel comprehensive answer evaluation paradigm that considers N potential answers in q 's answer space, denoted as $\{a_1^q, a_2^q, \dots, a_N^q\}$. First, LLM evaluates the trustworthiness of each (q, a_i^q) pair using the prompt p^e .

$$e_i = LLM(p^e(q, a_i^q)), i \in \{1, \dots, N\}. \quad (8)$$

Then, the obtained evaluations e_1, \dots, e_N can be integrated into existing paradigms to derive a more refined detection score for a by adjusting the prompts. For example, by adjusting the p^e for confidence calibration (*cf.* Eq 3) into a new prompt p^v , the detection score for a can be derived as,

$$c = Aggr(a, \{a_1, \dots, a_i\}), \quad (9)$$

where $a_i = LLM(p^v(q, e_1, \dots, e_N))$, $i \in \{1, \dots, D\}$.

The evaluations can also be integrated into self-evaluation by adjusting the prompt p^t in Eq 6.

Preliminary Experiments. We conduct a preliminary experiment to validate that considering more answers in the answer space to adjust the detection score is beneficial for self-detection.

Our hypothesis is that the evaluation of other answers can be leveraged to mitigate over-trust in the incorrect a . To demonstrate this, we employ counterfactual questions \bar{q} , which is minimally edited from q to have a different label within q 's answer space. We aim to utilize the label difference between q and \bar{q} to identify unreliable LLM-generated answer for q and adjust its detection score. Suppose the LLM-generated answers for \bar{q} and q are \bar{a} and a , respectively. If \bar{a} equals a , a and \bar{a} must have at least one wrong answer since \bar{q} and q have different labels. Thus the detection score of a on

$p(\bar{q})$ (denoted as c_a) should be reduced according to the detection score of \bar{a} on $p(\bar{q})$ (denoted as $c_{\bar{a}}$) because the increasing of $c_{\bar{a}}$ indicates the weakened c_a . Conversely, if \bar{a} differs from a , a and \bar{a} are relatively trustable, and c_a can be an average of itself and $c_{\bar{a}}$. Formally, c_a is re-calculated as

$$c = \begin{cases} \frac{1}{2}(c_a + c_{\bar{a}}) & \text{if } a \neq \bar{a}, \\ \frac{1}{2}(c_a + O(c_{\bar{a}})) & \text{else.} \end{cases} \quad (10)$$

where $O(c_{\bar{a}})$ denotes the detection score that \bar{q} 's label is not \bar{a} . In a k -classification task, we roughly estimate $O(c_{\bar{a}}) = \frac{1}{k-1}(1 - c_{\bar{a}})$.

We experiment with the CAD dataset (Kaushik et al., 2019), which contains human-annotated original and counterfactual data pairs for sentiment analysis (SA) and natural language inference (NLI) tasks. We compare the AUROC with self-consistency and Top- K verbalized methods to evaluate the self-detection performance on GPT-3.5 (see Section 5 and Appendix B for more details).

Figure 3 shows the AUROC and the statistics of detection scores for correct and incorrect q, a instances, respectively. We can observe that 1) the self-consistency and Top- K verbalized methods have notable over-trust. The detection scores for incorrect instances have large overlap with the correct ones, making it challenging to distinguish them. 2) Our strategy, denoted as *w/ cf*, improves AUROC by lowering detection scores on incorrect instances, showing that considering other answers can potentially alleviate the over-trust in incorrect answers. However, human-annotated counterfactual data is not easily available (Li et al., 2023b), motivating us to propose the following framework.

3 Think Twice Before Assure Framework

Implementing the proposed paradigm involves two key considerations. First, given the potential bias of LLM over-trust in the generated answer a , it is essential to develop strategies to resist this bias and thoroughly evaluate the trustworthiness of each answer a_i^q . Secondly, it is crucial to derive strategies to effectively combine these evaluations for effective self-detection of a . To address these concerns, we introduce the following two-step framework.

Step 1: Reflection and Justification. We first instruct LLM to reflect on the trustworthiness of each answer a_i^q and force LLM to seek justification for a_i^q as the correct answer of q , as defined by Eq. 8. The LLM is instructed with the prompt p^e in

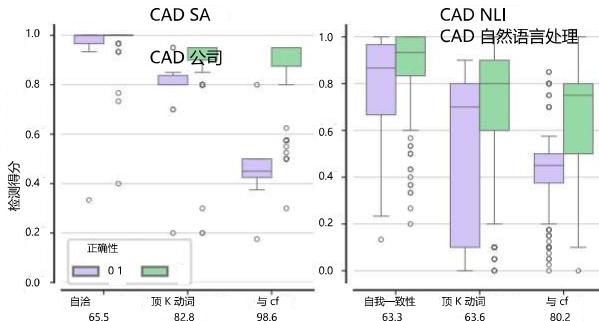


图 3: CAD 上自检测方法的比较。其中 “cf” 表示使用反事实数据的我们的策略。x 轴上显示的是 AUROC 值。左侧和右侧的方框分别代表错误和正确实例的检测分数。

在此背景下，我们提出了一种新颖的综合答案评估范式，该范式考虑了 q 的答案空间中的 N 个潜在答案，记作 $\{a, a, \dots, a\}$ 。首先，LLM 使用提示 p 评估每个 (q, a) 对的可靠性。

$$e = \text{LLM}(p(q, a)), i \in \{1, \dots, N\}. \quad (8)$$

然后，获得的评估值 e, \dots, e_{can} 可以整合到现有范式中去，通过调整提示来得到一个更精细的检测分数 a 。例如，通过将 p for 置信度校准（参看公式 3）调整到新的提示 p 中，可以推导出 a 的检测分数，如下所示：

$$c = \text{Aggr}(a, \{a, \dots, a\}), \quad (9) \text{ 其中 } a = \text{LLM}(p(q, e, \dots, e)), i \in \{1, \dots, D\}.$$

评估也可以通过调整提示针 Eq 6 的方式融入自我评估中。

初步实验。我们进行了一项初步实验。

一项实验旨在验证在答案空间中考虑更多答案以调整检测分数对自我检测有益。

我们的假设是，通过评估其他答案可以减轻对错误答案的过度信任。为了证明这一点，我们采用了反事实问题 $\neg q$ ，它是从 q 中经过最小编辑得到的，在 q 的答案空间中具有不同的标签。我们旨在利用 q 和 $\neg q$ 之间的标签差异来识别 LLM 生成的 q 的不可靠答案，并调整其检测分数。假设 LLM 生成的 $\neg q$ 和 q 的答案分别是：

$\neg a$ 和 a ，分别。如果 $\neg a$ 等于 a ，那么 a 和 $\neg a$ 至少有一个错误答案，因为 $\neg q$ 和 q 的标签不同。”

因此，在 $p(\neg q)$ 上的检测得分 c （表示为 c ）应根据在 $p(\neg q)$ 上的检测得分 a （表示为 c ）进行降低，因为 c 的增加意味着 c 的减弱。反之，如果 $\neg a$ 与 a 不同，那么 a 和 $\neg a$ 相对可信， c 可以是自身和 c 的平均值。形式上， c 重新计算如下：

$$c = \begin{cases} \frac{1}{2}(c + c) & \text{若 } a \neq \neg a, \\ \frac{1}{2}(c + O(c)) & \text{else.} \end{cases} \quad (10)$$

其中， $O(c)$ 表示 $\neg q$ 的标签不是 q 的检测得分。在 k 分类任务中，我们大致估计 $O(c) = 1 - c$ 。

我们实验了 CAD 数据集 (Kaushik 等人, 2019 年)，该数据集包含用于情感分析 (SA) 和自然语言推理 (NLI) 任务的人类标注的原始和反事实数据对。我们将 AUROC 与自治性和 Top-K 口头化方法进行比较，以评估 GPT-3.5 上的自我检测性能（更多细节请参见第 5 节和附录 B）。

图 3 展示了正确和错误 q, a 实例的 AUROC 和检测分数统计。我们可以观察到：1) 自治性和 Top-K 表达方法存在明显的过度信任问题。错误实例的检测分数与正确实例存在较大重叠，使得区分它们变得困难。2) 我们的策略，标记为 w/ cf，通过降低错误实例的检测分数，提高了 AUROC，这表明考虑其他答案可能有助于缓解对错误答案的过度信任。然而，人工标注的反事实数据不容易获得 (Li 等, 2023b)，这促使我们提出以下框架。

三思而后行，确保框架

实施所提出的范式涉及两个关键考虑因素。首先，鉴于 LLM 对生成的答案 a 过度信任的潜在偏差，开发抵制这种偏差的策略并彻底评估每个答案 a 的可信度至关重要。其次，推导出有效结合这些评估以实现 a 的自我检测的策略同样关键。为了解决这些问题，我们引入以下两步框架。

步骤 1：反思与论证。我们首先指导 LLM 反思每个答案的可信度，并强迫 LLM 为 q 的正确答案寻找依据，如公式 8 所定义。LLM 被提示使用提示符 pin 。

Table 1 to gather comprehensive evidence e_i from its knowledge, in order to support the rationality of using a_i^q to answer q . The rationality of this step is that p^e instructs LLM to abduct the justification from q and a_i^q , which avoids the LLM bias that lies in the generation direction from $p(q)$ to a . As a minor clue, generating CoT explanations from $p(q)$ before a has been validated to be ineffective for calibration (Zhang et al., 2023a).

| | |
|-------|--|
| p^e | The task is to [task description].
Question: $[q]$. Answer choices: $[a_1^q, \dots, a_N^q]$.
The answer is $[a_i^q]$.
Please generate an explanation to try to justify the answer judgment. |
| p^v | The task is to [task description].
Provide your N best guesses and the probability that each is correct (0.0 to 1.0) for the following question...
Question: $[q]$. Answer choices: $[a_1^q, \dots, a_N^q]$.
Possible explanation 1: $[e^1]$
...
Possible explanation n : $[e^N]$ |

Table 1: Prompts used in our T^3 framework. p^e prompts LLM to reflect and generate justification e_i for each a_i^q , and p^v prompts LLM to estimate confidence according to different e_i .

Step 2: Joint Confidence Calibration. After obtaining the justification e_i for each a_i^q , we choose to integrate these e_i with a confidence calibration method, the Top- K verbalized (*cf.* Eq. 5) to derive the confidence of answer a as the detection score. We choose this method due to its capability to generate a set of K potential answers and their respective probabilities efficiently in a single response, where we set K as the number of answers N . As indicated in the prompt p^v of Table 1, the generated justifications e_i can be seamlessly integrated for confidence calibration of Top- K verbalized.

An alternative approach to determine the final detection score is to put one justification to each p^v , generating N distinct confidence scores for answer a , and then compute the averaged confidence score as the detection score. We do not choose this setting as prompting LLM to estimate from different perspectives via a unified prompt is more efficient and effective than a simple average of the confidence scores (further validated in Section 5.2). Moreover, we find that the detection scores are sensitive to the order of justification in p^v , thus we shuffle the order of e^i in p^v and compute the averaged score. Notably, the T^3 framework can be combined with existing approaches, such as prompt ensemble (Jiang et al., 2023), and Hybrid method

which adjust the detection score based on the difference with other methods (Xiong et al., 2023).

4 Related Work

Confidence Calibration of LLM. Confidence calibration has been previously studies in neural networks (Guo et al., 2017) and applied in NLP models (Desai and Durrett, 2020; Dan and Roth, 2021; Hu et al., 2023). After the advent of LLM, many confidence calibration methods utilize the output token probability, such as semantic uncertainty (Kuhn et al., 2023), temperature scaling (Shih et al., 2023), entropy-based (Huang et al., 2023c), semantic significance (Duan et al., 2023), and fine-tuning for calibration (Jiang et al., 2021; Lin et al., 2022a). Zhang et al. (2023b) also employ model ensemble for calibration. Our research is orthogonal to them, since we focus on black-box API LLM itself. Other recent work suitable for black-box LLM includes fidelity elicitation (Zhang et al., 2024), fact elicitation (Zhao et al., 2024) and perturbation generation (Gao et al., 2024).

Self-Evaluation of LLM. LLM self-evaluation often focuses on specific domains, *e.g.*, code generation (Zhou et al., 2023), natural language generation (Lin et al., 2023) and fact checking (Manakul et al., 2023). The general methods include P(True) (Kadavath et al., 2022) and directly asking LLM (Li et al., 2024b). Feng et al. (2024) also performs answer reflection and employs model collaboration, yet they still focus on answers generated by LLM. Note that self-detect (Zhao et al., 2023c) is also a general self-evaluation method following the three-step confidence calibration pipeline.

Other works that are related but orthogonal to us include training independent models for LLM evaluation (Wang and Li, 2023; Li et al., 2023c; Khalifa et al., 2023; Zhao et al., 2023b; Li et al., 2024a), and using external tools for LLM verification (Min et al., 2023; Ni et al., 2023). They are usually applied to specific domains, while we aim at LLM self-detection for general tasks. Also, fine-tuning LLM for better trustworthiness (An et al., 2023; Tian et al., 2023a) is orthogonal to us.

Application of LLM Self-Detection. The outcome of self-detection can be applied in many ways to avoid hallucination and erroneous outputs, such as identifying potentially hallucinated generation for knowledge retrieval and verification (Zhao et al., 2023a), guided output decoding (Xie et al., 2023),

“LLM被指示使用提示 “将表 1 固定” 来收集其知识中的全面证据，以支持使用 a 回答 q 的合理性。这一步骤的合理性在于，它指示LLM从 q 和 a 中归纳出论证，从而避免了从 $p(q)$ 到 a 的生成方向中存在的LLM偏差。作为一个小线索，在验证 a 之前从 $p(q)$ 生成 CoT 解释已被证明对于校准 (Zhang et al., 2023a) 是无效的。”

| | |
|----------|---|
| <p>p</p> | <p>任务是要[任务描述]。
问题: [q]。选项: [a, ..., a]。
答案为[a]。
请提供解释以尝试证明答案判断的合理性。</p> |
| <p>p</p> | <p>任务是要[任务描述]。
请提供您对以下问题的 N 个最佳猜测，并给出每个猜测正确的概率 (0.0 至 1.0)。
问题: [q]。选项: [a, ..., a]。可能解释 1: [e]
... 可能解释 n: [e]</p> |

表 1: T 框架中使用的提示。pprompts LLM 用于对每个 a 进行反思和生成理由，以及 pprompts LLM 用于根据不同的 e 估计置信度。

步骤 2: 联合置信度校准。在...

在为每个 a 提供理由的基础上，我们选择将这些理由与置信度校准方法相结合，即 Top-K 语言化 (参见公式 5)，以推导出答案 a 的置信度作为检测分数。我们选择这种方法是因为它能够在单个响应中高效地生成一组 K 个潜在答案及其相应的概率，我们将 K 设置为答案数量 N 。如提示表 1 所示，生成的理由可以无缝集成到 Top-K 语言化的置信度校准中。

一种确定最终检测分数的替代方法是针对每个 p 提出一个论据，为答案 a 生成 N 个不同的置信度分数，然后计算置信度分数的平均值作为检测分数。我们并未选择这种设置，因为通过统一的提示让 LLM 从不同角度进行估计比简单平均置信度分数更高效、更有效 (这在第 5.2 节中得到进一步验证)。此外，我们发现检测分数对 p 中论据的顺序敏感，因此我们打乱 p 中论据的顺序并计算平均分数。值得注意的是，T 框架可以与现有方法相结合，例如提示集成 (Jiang 等人, 2023)。

混合方法，该方法根据与其他方法的差异调整检测得分 (Xiong 等, 2023)。

4 相关工作

”

LLM的置信度校准。置信度校准之前已在神经网络 (Guo 等, 2017) 中进行过研究，并在 NLP 模型中得到了应用 (Desai 和 Durrett, 2020; Dan 和 Roth, 2021; Hu 等, 2023)。LLM出现之后，许多置信度校准方法开始利用输出标记概率，例如语义不确定性 (Kuhn 等, 2023)、温度缩放 (Shih 等, 2023)、基于熵的 (Huang 等, 2023c)、语义重要性 (Duan 等, 2023) 以及校准的微调 (Jiang 等, 2021; Lin 等, 2022a)。Zhang 等 (2023b) 还采用了模型集成进行校准。我们的研究与他们正交，因为我们专注于黑盒 API LLM 本身。其他适用于黑盒LLM的近期工作包括保真度提取 (Zhang 等, 2024)、事实提取 (Zhao 等, 2024) 和扰动生成 (Gao 等, 2024)。

”

自我评估: LLM。LLM的自我评估

经常关注特定领域，例如代码生成 (周等, 2023 年)、自然语言生成 (林等, 2023 年) 和事实核查 (曼库尔等, 2023 年)。通用方法包括 $P(\text{True})$ (卡达瓦特等, 2022 年) 和直接询问LLM (李等, 2024b)。冯等 (2024 年) 还进行答案反思并采用模型协作，但他们仍然关注由LLM生成的答案。请注意，自检 (赵等, 2023c) 也是一种遵循三步置信度校准流程的通用自我评估方法。

其他与此相关但与我们的研究方向正交的工作包括为LLM评估训练独立的模型 (王和李, 2023; 李等, 2023c; 哈利法等, 2023; 赵等, 2023b; 李等, 2024a)，以及使用外部工具进行LLM验证 (闵等, 2023; 倪等, 2023)。这些方法通常应用于特定领域，而我们旨在实现LLM通用任务的自检测。此外，为了提高可信度对LLM进行微调 (安等, 2023; 田等, 2023a) 也与我们的研究正交。

应用LLM 自检。输出-

自我检测技术可以应用于多种方式以避免幻觉和错误输出，例如用于识别可能存在的幻觉生成以进行知识检索和验证 (赵等, 2023a)，以及指导输出解码 (谢等, 2023)。

identifying ambiguous questions (Hou et al., 2023), selective generation (Ren et al., 2023a; Zablotskaia et al., 2023), and LLM self-improve (Huang et al., 2023a). More applications can be found in this survey (Pan et al., 2023).

Generating Multiple-Side Rationale. Some existing research on LLM also utilizes multiple-side rationales similar to T^3 , such as Process of Elimination (Balepur et al., 2023), Maieutic Prompting (Jung et al., 2022), and Debate (Michael et al., 2023). Our T^3 differs from these work mainly in that T^3 incorporates multiple side rationales into LLM self-detection, by combining the rationales with Top- K Verbalized confidence estimation with designed prompts. Besides, the format of multi-side rationale of T^3 is only similar to Jung et al. (2022), while T^3 can handle a wider range of tasks.

5 Experiments

Setup. We conduct experiments on six datasets across three tasks. IMDB (Maas et al., 2011) and Flipkart (Vaghani and Thummar, 2023) for SA, SNLI (Bowman et al., 2015) and HANS (McCoy et al., 2019) for NLI, CommonsenseQA (Talmor et al., 2019) and PIQA (Bisk et al., 2020) for commonsense question answering (CQA). For LLMs, we utilize GPT-3.5 (*gpt-3.5-turbo-1106*) from OpenAI³, GLM-4 (Du et al., 2022) from ZhipuAI⁴, and Gemini (*gemini-1.0-pro-001*) from Google⁵. Dataset statistics and LLM hyperparameters are listed in Appendices A.1 and A.2.

Compared Methods. We utilize the following categories of compared methods. For the first paradigm, we include **Self-cons** (Wang et al., 2022) (*cf.* Eq. 4), **CoT-cons**, an extension of Self-cons by instructing LLM to output the CoT reasoning before the answer, **Induced-cons** (Xiong et al., 2023), adding different misleading contexts to the input to examine consistency. **Top- K Verb** (Tian et al., 2023b) (*cf.* Eq. 5), **Hybrid** (Xiong et al., 2023), an integration of Top- K Verb and Self-cons/CoT-cons, where we show the better results, **Self-detect** (Zhao et al., 2023c), taking the answer entropy of multiple rephrased questions, and **CAPE** (Jiang et al., 2023), a prompt ensemble method that we implement on Top- K Verb. For the second paradigm, we utilize the general **P(True)** (Kadavath et al., 2022), and

Self-Probing (Xiong et al., 2023), directly asking LLM to explain and evaluate the confidence of the given answer. Finally, to show the flexibility of T^3 in combining with existing methods to further improve self-detection, we show the performance of Hybrid T^3 with Top- K Verb ($T^3 + \text{Top-}K \text{ Verb}$), and T^3 with prompt ensemble following CAPE ($T^3 + \text{PE}$). For a fair comparison, we generate the target answer for each dataset with LLM temperature as 0, and compare all methods based on this target answer (*cf.* Eq 1). More details are in Appendices A.3 and A.4, including a comparison on the number of API calls showing T^3 's reasonable cost.

Evaluation Metrics. We mainly use **AUROC** (Boyd et al., 2013) and **PRAUC** (Manning and Schutze, 1999) to evaluate the self-detection ability. They assess the effectiveness of detection scores in distinguishing answer correctness using true positive/false positive and precision/recall curves, respectively. Additionally, we use the Expected Calibration Error (**ECE**) to evaluate the calibration performance for confidence calibration methods.

5.1 Results

Table 2 shows the performance of the compared methods on GPT-3.5. We can observe the followings. 1) T^3 outperforms all compared methods in AUROC and PRAUC on all datasets except HANS and PIQA, and in ECE on all datasets except SNLI, demonstrating its effectiveness. 2) After combining T^3 with other methods *i.e.*, Top- K Verb and PE, our method surpasses all compared methods on all datasets in the three evaluation metrics, showing the potential and flexibility of T^3 in combining with others to further improve self-detection. 3) Hybrid with Top- K Verb usually improves T^3 's performance in AUROC and PRAUC, which is in line with the performance improvement from Self-cons/CoT-cons to Hybrid. 4) CAPE is very competitive in AUROC and PRAUC, showing that the self-detection is largely influenced by the prompt. Combining T^3 with PE usually improves T^3 in AUROC and PRAUC except for SNLI and Flipkart, which is in line with the performance decrease from Top- K Verb to CAPE. This is potentially related to the prompt sensitivity of these methods and the specific prompts adopted. 5) For other methods, CoT-cons outperforms Self-cons in AUROC and PRAUC in 5 out of 6 datasets, as many tasks performs better with CoT reasoning. P(True) has

³<https://openai.com/blog/openai-api>.

⁴<https://open.bigmodel.cn/>.

⁵<https://gemini.google.com/app>.

识别模糊性问题 (Hou 等人, 2023 年), 选择性生成 (Ren 等人, 2023a; Zablotskaia 等人, 2023) , 以及LLM自我提升 (Huang 等人, 2023a) 。更多应用可以在本调查中找到 (Pan 等人, 2023) 。

生成多角度的论证。一些例子——

现有关于LLM的研究也采用了类似于 T 的多角度论证方法, 例如排除法 (Balepur 等人, 2023 年) 、助产式提示 (Jung 等人, 2022 年) 和辩论

(Michael 等人, 2023 年) 。我们的 T 与这些工作主要区别在于, T 将多角度论证融入LLM的自检测中, 通过结合论证与设计的提示进行 Top-K 语言化置信度估计。此外, T 的多角度论证格式仅与 Jung 等人 (2022 年) 相似, 而 T 可以处理更广泛的任務。

自我探测 (熊等, 2023) , 直接要求LLM解释并评估给定答案的置信度。最后, 为了展示 T 在结合现有方法以进一步改进自我检测方面的灵活性, 我们展示了混合 T 与 Top-K 动词 (T + Top-K 动词) 以及 T 与遵循 CAPE 的提示集成 (T + PE) 的性能。为了公平比较, 我们使用LLM温度为 0 生成每个数据集的目标答案, 并基于此目标答案比较所有方法 (参见图 1) 。更多细节请见附录 A.3 和 A.4, 包括关于 API 调用次数的比较, 展示了 T 的合理成本。

评估指标。我们主要使用 AUROC (Boyd 等人, 2013 年) 和 PRAUC (Manning 和 Schutze, 1999 年) 来评估自检测能力。它们分别通过真正例/假正例和精确率/召回率曲线来评估检测分数在区分答案正确性方面的有效性。此外, 我们还使用期望校准误差 (ECE) 来评估置信度校准方法的校准性能。

5 实验研究

设置。我们在三个任务上对六个数据集进行实验。

IMDB (Maas 等人, 2011 年) 和 Flipkart

(Vaghani 和 Thummar, 2023 年) 用于句子相似度 (SA) , SNLI (Bowman 等人, 2015 年) 和 HANS (McCoy 等人, 2019 年) 用于自然语言推理 (NLI) , CommonsenseQA (Talmor 等人, 2019 年) 和 PIQA (Bisk 等人, 2020 年) 用于常识问答 (CQA) 。对于LLMs, 我们使用了 OpenAI 的 GPT-3.5 (gpt-3.5-turbo-1106) , ZhipuAI 的 GLM-4 (Du 等人, 2022 年) 以及 Google 的 Gemini (gemini-1.0-pro-001) 。数据集统计和LLM的超参数列于附录 A.1 和 A.2。

比较方法。我们采用了以下几类比较方法。对于第一个范式, 我们包括 Self-cons (王等, 2022 年)

(参见图 4) , CoT-cons, Self-cons 的扩展, 通过指示LLM在给出答案之前输出 CoT 推理, Induced-cons (熊等, 2023 年) , 向输入添加不同的误导性上下文以检验一致性。Top-K Verb (田等, 2023b) (参见图 5) , Hybrid (熊等, 2023 年) , Top-K Verb 和 Self-cons/CoT-cons 的集成, 其中我们展示了更好的结果, Self-detect

(赵等, 2023c) , 通过多个改写问题的答案熵进行检测, 以及 CAPE (江等, 2023 年) , 一种我们在 Top-K Verb 上实现的提示集成方法。对于第二个范式, 我们采用了一般的 P(True) (卡达瓦特等, 2022 年) 。

5.1 结果

表 2 展示了在 GPT-3.5 上比较方法的性能。我们可以观察到以下几点: 1) T 在所有数据集上 (除了 HANS 和 PIQA) 的 AUROC 和 PRAUC 以及所有数据集上 (除了 SNLI) 的 ECE 方面均优于所有比较方法, 证明了其有效性。2) 将 T 与其他方法结合, 例如 Top-K Verb 和 PE, 我们的方法在所有数据集上的三个评估指标上都超越了所有比较方法, 显示了 T 与其他方法结合以进一步提高自我检测的潜力和灵活性。3) 与 Top-K Verb 结合的混合方法通常可以提高 T 在 AUROC 和 PRAUC 方面的性能, 这与从 Selfcons/CoT-cons 到混合的性能提升相一致。4) CAPE 在 AUROC 和 PRAUC 方面具有很强的竞争性, 表明自我检测在很大程度上受到提示的影响。将 T 与 PE 结合通常可以提高 T 在 AUROC 和 PRAUC 方面的性能, 除了 SNLI 和 Flipkart, 这与从 Top-K Verb 到 CAPE 的性能下降相一致。这可能与这些方法的提示敏感性和采用的特定提示有关。5) 对于其他方法, CoT-一致性在 6 个数据集中有 5 个在 AUROC 和 PRAUC 方面优于自一致性, 因为许多任务在 CoT 推理下表现更佳。P(真)有

——<https://openai.com/blog/openai-api>。

<https://open.bigmodel.cn/>

<https://gemini.google.com/app>。

| | IMDB | | | Flipkart | | |
|--------------|-------------|-------------|--------------|-------------|-------------|--------------|
| | AUROC ↑ | PRAUC ↑ | ECE ↓ | AUROC ↑ | PRAUC ↑ | ECE ↓ |
| Self-cons | 65.5 | 96.8 | 0.115 | 71.4 | 91.4 | 0.106 |
| Cot-cons | 75.6 | 97.7 | 0.104 | 72.8 | 91.9 | 0.133 |
| Top-K Verb | 82.8 | 98.5 | 0.242 | 79.3 | 93.7 | 0.131 |
| P(True) | 80.1 | 98.1 | 0.104 | 54.5 | 86.7 | 0.243 |
| Hybrid | 87.0 | <u>98.8</u> | 0.183 | 79.5 | 94.2 | 0.176 |
| Self-detect | 68.9 | 97.1 | 0.320 | 71.2 | 91.4 | 0.146 |
| CAPE | 87.7 | 98.9 | 0.096 | 76.4 | 93.9 | 0.107 |
| T^3 | 87.9 | 98.9 | 0.045 | <u>81.3</u> | <u>94.5</u> | 0.015 |
| + Top-K Verb | <u>88.0</u> | 98.9 | <u>0.052</u> | 81.6 | 94.9 | <u>0.019</u> |
| + PE | 88.1 | 98.9 | 0.056 | 74.2 | 92.9 | 0.157 |

(a) SA.

| | SNLI | | | HANS | | |
|--------------|-------------|-------------|--------------|-------------|-------------|--------------|
| | AUROC ↑ | PRAUC ↑ | ECE ↓ | AUROC ↑ | PRAUC ↑ | ECE ↓ |
| Self-cons | 63.3 | 71.4 | 0.047 | 56.0 | 64.8 | 0.051 |
| Cot-cons | 66.7 | 73.8 | 0.043 | 59.4 | 67.9 | 0.152 |
| Top-K Verb | 63.6 | 74.0 | 0.089 | 53.3 | 64.9 | 0.273 |
| P(True) | 55.4 | 67.4 | 0.117 | 60.8 | 70.1 | 0.067 |
| Hybrid | 66.7 | 78.8 | <u>0.029</u> | 62.0 | 71.1 | 0.193 |
| Self-detect | 59.3 | 68.5 | 0.142 | 55.3 | 64.5 | 0.063 |
| CAPE | 69.0 | 79.6 | 0.030 | 71.9 | <u>80.1</u> | <u>0.028</u> |
| T^3 | 77.9 | <u>84.6</u> | 0.157 | 69.9 | 77.5 | 0.022 |
| + Top-K Verb | <u>77.1</u> | 84.7 | 0.024 | 71.3 | 79.6 | 0.030 |
| + PE | 70.8 | 76.7 | 0.130 | <u>74.5</u> | 81.2 | 0.034 |

(b) NLI.

| | CommonsenseQA | | | PIQA | | |
|--------------|---------------|-------------|--------------|-------------|-------------|--------------|
| | AUROC ↑ | PRAUC ↑ | ECE ↓ | AUROC ↑ | PRAUC ↑ | ECE ↓ |
| Self-cons | 70.7 | 81.7 | 0.151 | 78.6 | 94.0 | 0.043 |
| Cot-cons | 81.8 | 88.9 | 0.049 | 76.7 | 94.2 | 0.097 |
| Top-K Verb | 69.4 | 81.5 | 0.026 | 76.8 | 93.3 | 0.060 |
| P(True) | 62.5 | 78.0 | 0.097 | 71.9 | 93.9 | 0.176 |
| Hybrid | 77.5 | 89.0 | <u>0.015</u> | 82.4 | 95.5 | 0.088 |
| Self-detect | 67.9 | 81.5 | 0.261 | 68.5 | 91.0 | 0.161 |
| CAPE | 78.7 | 88.8 | 0.021 | <u>87.9</u> | <u>97.8</u> | 0.067 |
| T^3 | 83.5 | 90.7 | 0.009 | 83.4 | 95.2 | <u>0.016</u> |
| + Top-K Verb | 85.8 | <u>93.4</u> | 0.017 | 85.3 | 96.2 | 0.010 |
| + PE | <u>84.4</u> | <u>92.1</u> | 0.019 | <u>90.3</u> | <u>97.9</u> | 0.034 |

(c) CQA.

Table 2: Results of the compared methods on GPT-3.5. Bold font and underline indicate the best and second best performance, respectively.

ambivalent results which limits its applicability.

5.2 In-depth Analysis

Ablation Studies. We conduct the following ablation studies to further validate the rationality of our framework design. 1) *w/ CoT expl*: substituting e_1, \dots, e^N in p^v with N different CoT reasoning generated from $p(q)$ to reveal the rationality of reflection on various answers. 2) *sep expl*: placing a single e_i in p^v each time and calculating the averaged detection score to reveal the effectiveness of joint considering all e_i in one p^v . 3) *w/o shuffle*: ablating the order shuffling of e_i in p^v .

From Table 3, we can observe that: 1) *w/ CoT expl* largely underperforms T^3 on all three tasks, demonstrating the rationality of pushing LLM to reflect and justify from each answer’s perspective. 2) *sep expl* underperforms T^3 on both SA and NLI tasks, showing that jointly considering multiple justifications in one prompt is often more benefi-

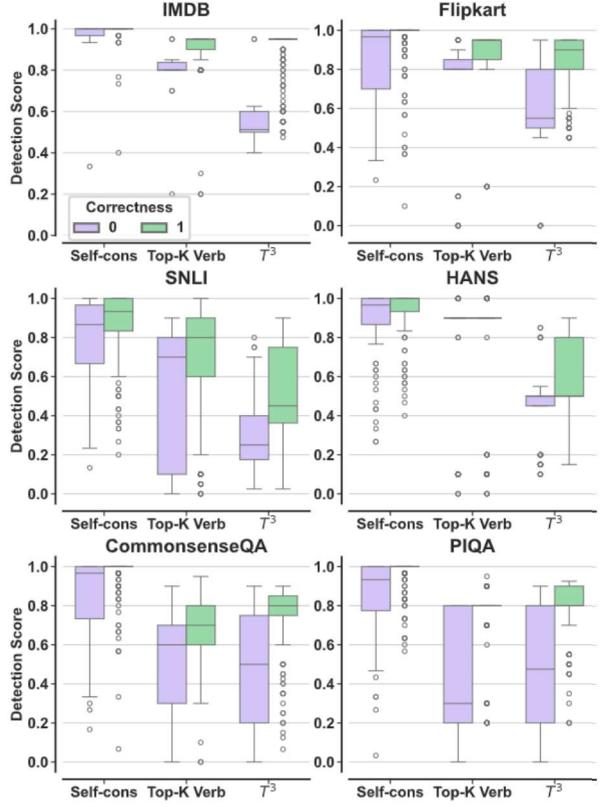


Figure 4: Visualization of bias mitigation effect of T^3 which largely reduces the detection score overlaps between correct (right) and incorrect (left) instances.

cial, and thus we choose this setting. It slightly outperforms T^3 on the CQA task, potentially due to the higher independency and objectivity of the answer choices. 3) *w/o shuffle* also underperforms T^3 , indicating that there exists order sensitivity for e_i . Order shuffling and score average improve self-detection by mitigating their position bias.

Effect on Bias Mitigation. Since our goal of improving self-detection is to reduce the over-trust on incorrect answers, we show the statistics of the detection scores for each dataset regarding the answer correctness in Figure 4 to reveal the mechanism of T^3 . We compare T^3 with Self-cons and Top-K Verb which are witnessed with over-trust. We can observe that T^3 clearly reduces the detection score overlaps between correct and incorrect q, a instances on all datasets, and significantly decreases the detection scores on incorrect instances in IMDB, Flipkart, SNLI and HANS. Thus, the answer accuracy is more separable by the detection score, achieving better self-detection.

Effect on Selective Prediction via Detection Score. To show the utility of the detection score, we conduct experiments in selective prediction.

| | IMDB | Flipkart | AUROC \downarrow | PRAUC \downarrow | ECE \downarrow | AUROC \downarrow | PRAUC \downarrow | ECE \downarrow |
|--|------|----------|--------------------|--------------------|------------------|--------------------|--------------------|------------------|
| <hr/> | | | | | | | | |
| 自我一致性 65.5 96.8 0.115 71.4 91.4 0.106 主题一致性 75.6 97.7 0.104 72.8 91.9
0.133 Top-K 动词 82.8 98.5 0.242 79.3 93.7 0.131 P(True) 80.1 98.1 0.104 54.5 86.7
0.243 混合 87.0 98.8 0.183 79.5 94.2 0.176 自检测 68.9 97.1 0.320 71.2 91.4 0.146
CAPE 87.7 98.9 0.096 76.4 93.9 0.107 T 87.9 98.9 0.045 81.3 94.5 0.015 + Top-K 动词
88.0 98.9 0.052 81.6 94.9 0.019 + PE 88.1 98.9 0.056 74.2 92.9 0.157 | | | | | | | | |
| <hr/> | | | | | | | | |
| (一) SA。 | | | | | | | | |
| <hr/> | | | | | | | | |
| SNLI HANS AUROC \downarrow PRAUC \downarrow ECE \downarrow AUROC \downarrow PRAUC \downarrow ECE \downarrow 自检 63.3 71.4 0.047 56.0
64.8 0.051 CoT-cons 66.7 73.8 0.043 59.4 67.9 0.152 Top-K Verb 63.6 74.0 0.069 53.3
64.9 0.273 P(True) 55.4 67.4 0.117 60.8 70.1 0.067 混合 66.7 78.8 0.029 62.0 71.1
0.193 自检 59.3 68.5 0.142 55.3 64.5 0.063 CAPE 69.0 79.6 0.030 71.9 80.1 0.028 T
77.9 84.6 0.157 69.9 77.5 0.022 + Top-K Verb 77.1 84.7 0.024 71.3 79.6 0.030 + PE
70.8 76.7 0.130 74.5 81.2 0.034 | | | | | | | | |
| <hr/> | | | | | | | | |
| (b) NLI。 | | | | | | | | |
| <hr/> | | | | | | | | |
| 常识问答 PIQA AUROC \downarrow PRAUC \downarrow ECE \downarrow AUROC \downarrow PRAUC \downarrow ECE \downarrow | | | | | | | | |
| <hr/> | | | | | | | | |
| ***
Self-cons 70.7 81.7 0.151 78.6 94.0 0.043 CoT-cons 81.8 88.9 0.049 76.7 94.2 0.097
Top-K Verb 69.4 81.5 0.026 76.8 93.3 0.060 P(True) 62.5 78.0 0.097 71.9 93.9 0.176
Hybrid 77.5 89.0 0.015 82.4 95.5 0.088 Self-detect 67.9 81.5 0.261 68.5 91.0 0.161
CAPE 78.7 88.8 0.021 87.9 97.8 0.067 T 83.5 90.7 0.009 83.4 95.2 0.016 + Top-K
Verb 85.8 93.4 0.017 85.3 96.2 0.010 + PE 84.4 92.1 0.019 90.3 97.9 0.034 | | | | | | | | |
| <hr/> | | | | | | | | |
| Self-cons 70.7 81.7 0.151 78.6 94.0 0.043 CoT-cons 81.8 88.9 0.049 76.7 94.2 0.097
Top-K Verb 69.4 81.5 0.026 76.8 93.3 0.060 P(True) 62.5 78.0 0.097 71.9 93.9 0.176
Hybrid 77.5 89.0 0.015 82.4 95.5 0.088 Self-detect 67.9 81.5 0.261 68.5 91.0 0.161
CAPE 78.7 88.8 0.021 87.9 97.8 0.067 T 83.5 90.7 0.009 83.4 95.2 0.016 + Top-K
Verb 85.8 93.4 0.017 85.3 96.2 0.010 + PE 84.4 92.1 0.019 90.3 97.9 0.034 | | | | | | | | |
| <hr/> | | | | | | | | |
| (c) CQA 版权所有。 | | | | | | | | |

表 2: 比较方法在 GPT-3.5 上的结果。粗体和下划线分别表示最佳和次佳性能。

矛盾的结果限制了其适用性。

5.2 深入分析

消融实验研究。我们进行了以下消融实验，以进一步验证框架设计的合理性。1) 带 CoT 解释：将 $e, \dots, ein p$ 替换为从 $p(q)$ 生成的 N 个不同的 CoT 推理，以揭示对各种答案进行反思的合理性。2) 分离解释：每次只放置一个 $ein peach$ ，并计算平均检测分数，以揭示联合考虑所有 $ein one p$ 的有效性。3) 无洗牌：消融 p 中 ein 的顺序洗牌。

从表 3 中我们可以观察到：1) 带有 CoT 解释的模型在所有三个任务上都明显低于 T 模型，这证明了将 LLM 推向从每个答案的角度进行反思和论证的合理性。2) 分离解释在 SA 和 NLI 任务上都低于 T 模型，这表明在一个提示中联合考虑多个论证通常更有益。

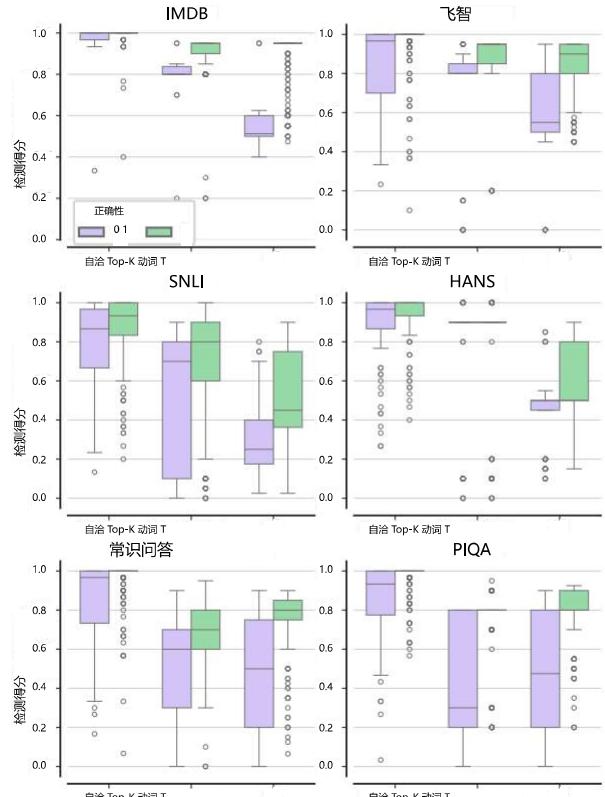


图 4: T 方法对偏差缓解效果的可视化，该方法在很大程度上减少了正确(右侧)和错误(左侧)实例之间的检测得分重叠。

我们选择了这种设置，因为它在 CQA 任务上略优于 T，这可能是由于答案选项的独立性和客观性更高。3) 不进行洗牌的情况下也表现不佳，这表明存在对 e 的顺序敏感性。通过缓解位置偏差，顺序洗牌和得分平均提高了自我检测能力。

效果对偏差缓解的影响。鉴于我们的目标是——为了减少对错误答案的过度信任，我们展示了每个数据集关于答案正确性的检测分数统计，如图 4 所示，以揭示 T 的机制。我们将 T 与 Self-cons 和 Top-K Verb 进行了比较，后者都存在过度信任的问题。我们可以观察到，T 在所有数据集上明显减少了正确和错误 q, a 实例之间的检测分数重叠，并在 IMDB、Flipkart、SNLI 和 HANS 上显著降低了错误实例的检测分数。因此，通过检测分数区分答案的准确性更高，实现了更好的自我检测。

《通过检测的预测选择效应》

得分。为了展示检测得分的实用性，我们进行了选择性预测实验。

| | IMDB | | Flipkart | | SNLI | | HANS | | CommonsenseQA | | PIQA | |
|-------------|--------|--------|----------|--------|---------|---------|--------|--------|---------------|--------|---------|---------|
| | AUROC↑ | PRAUC↑ | AUROC↑ | PRAUC↑ | AUROC ↑ | PRAUC ↑ | AUROC↑ | PRAUC↑ | AUROC↑ | PRAUC↑ | AUROC ↑ | PRAUC ↑ |
| T^3 | 87.9 | 98.9 | 81.3 | 94.5 | 77.9 | 84.6 | 69.9 | 77.5 | 83.5 | 90.7 | 83.4 | 95.2 |
| w/ CoT expl | 72.4 | 97.5 | 76.6 | 93.4 | 67.1 | 75.2 | 53.7 | 64.1 | 78.7 | 86.8 | 81.3 | 94.8 |
| sep expl | 86.5 | 98.8 | 79.5 | 94.2 | 68.5 | 75.3 | 54.1 | 63.8 | 80.3 | 87.8 | 84.0 | 95.8 |
| w/o shuffle | 75.9 | 98.3 | 71.7 | 92.0 | 70.6 | 77.6 | 60.7 | 67.9 | 83.3 | 92.0 | 80.4 | 94.3 |

Table 3: Ablation studies.

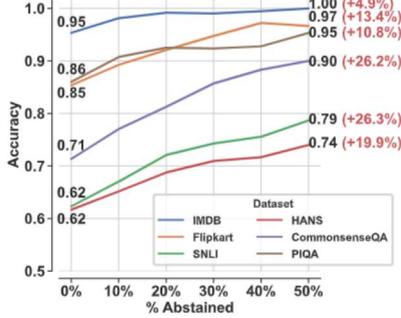


Figure 5: Accuracy improvement of selective prediction on T^3 detection scores.

The idea of selective prediction is to abstain the LLM-generated answers with low detection score to maintain better accuracy of the remaining instances. In Figure 5, we show the accuracy of the remaining instances by abstaining 0% - 50% of instances with the lowest detection scores from T^3 . We can observe that by increasing the percentage of abstained instances, the accuracy for these datasets gradually improves around 10% - 30%, and IMDB even achieves 100% accuracy. Naturally, the increase for datasets with lower accuracy is generally easier than datasets with higher accuracy. The result shows that T^3 possess strong potential to be applied in selective prediction scenarios.

Analysis on the Robustness of T^3 . We evaluate the robustness of T^3 from three aspects: different target answers, different LLMs, and parameter sensitivity. In addition, we examine prompt sensitivity of p^e and p^v in Appendix C.

Firstly, the generation of target answer a may vary under LLM randomness, *e.g.*, setting the temperature greater than 0. We verify the robustness of T^3 by utilizing **different target answers**, *i.e.*, the majority answer of Self-cons (a^{sc}) and CoT-cons (a^{cc}), respectively, as shown in Table 4. We can observe the following. 1) For both sets of target answers, T^3 largely outperforms baselines, showing its effectiveness. 2) Different target answers may have very different self-detection performance. Specifically, a^{cc} on CommonsenseQA has a sharp decrease in AUROC of T^3 and CoT-cons compared with the other target answers, which is potentially due to the majority voting with CoT explanation

| | Flipkart | HANS | CommonsenseQA | | |
|----------|------------|-------------|---------------|-------------|--------|
| | | | | AUROC↑ | PRAUC↑ |
| a^{sc} | Self-cons | 72.7 | 52.7 | 68.2 | |
| | CoT-cons | 74.4 | 57.5 | 80.4 | |
| | Top-K Verb | 80.4 | 51.8 | 69.2 | |
| | T^3 | 82.2 | 69.5 | 82.7 | |
| a^{cc} | Self-cons | 78.3 | 57.0 | 68.1 | |
| | CoT-cons | 79.2 | 57.8 | 74.3 | |
| | Top-K Verb | 83.9 | 53.3 | 67.5 | |
| | T^3 | 84.3 | 69.2 | 75.0 | |

Table 4: AUROC on two different target answers.

| | Flipkart | | HANS | | CommonsenseQA | |
|--------------|-------------|-------------|-------------|-------------|---------------|-------------|
| | AUROC↑ | PRAUC↑ | AUROC↑ | PRAUC↑ | AUROC ↑ | PRAUC ↑ |
| CoT Cons | 73.4 | 88.8 | 66.4 | 87.5 | 83.1 | 97.0 |
| Top-K Verb | 81.1 | 92.1 | 65.4 | 88.0 | 72.3 | 95.3 |
| Hybrid | 80.4 | 92.0 | 69.9 | 89.4 | 79.4 | 97.2 |
| CAPE | 82.3 | 92.7 | <u>82.4</u> | <u>94.0</u> | <u>80.0</u> | 96.8 |
| T^3 | <u>83.3</u> | 93.4 | 82.0 | 93.9 | 72.5 | 96.0 |
| + Top-K Verb | 82.7 | 93.2 | 80.9 | 93.9 | 81.0 | 97.6 |
| + PE | 83.8 | 93.4 | 84.9 | 95.7 | 76.9 | 96.6 |

Table 5: Performance comparison of Flipkart, HANS and CommonsenseQA on GLM-4.

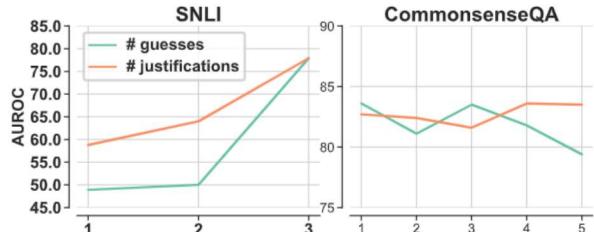


Figure 6: Parameter sensitivity, *i.e.*, changing the number of justifications and number of guesses in p^v .

diminished the effect of the explanations in T^3 .

Secondly, we evaluate T^3 on **different LLMs**. Table 5 shows the performance comparison of Flipkart, HANS and CommonsenseQA on GLM-4. We can observe that across different LLMs, combining T^3 with PE or Top-K Verb outperforms compared methods, validating its effectiveness. Besides, the self-detection ability may vary greatly across LLMs, *e.g.*, T^3 's AUROC of HANS on GLM-4 largely outperforms that on GPT-3.5. More results on Gemini can be found in Appendix D.

Thirdly, we evaluate the **parameter sensitivity** of T^3 by changing the number of justifications and number of guesses in p^v . We conduct experiments on CommonsenseQA with five answer choices, and SNLI with three answer choices. From Figure 6, we can observe the followings. 1) A larger number of justifications increases the performance on both datasets, indicating a sufficient number of justifica-

| | IMDB | Flipkart | SNLI | HANS | CommonsenseQA | PIQA | AUROC↑ | PRAUC↑ | AUROC↑ | PRAUC↑ | AUROC↑ | PRAUC↑ | AUROC↑ | PRAUC↑ | | | | | | | | | | | | | | | |
|------|------|----------|------|------|---------------|------|--------|--------|--------|--------|--------|--------|----------|--------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| T | 87.9 | 98.9 | 81.3 | 94.5 | 77.9 | 84.6 | 69.9 | 77.5 | 83.5 | 90.7 | 83.4 | 95.2 | 带 CoT 解释 | 72.4 | 97.5 | 76.6 | 93.4 | 67.1 | 75.2 | 53.7 | 64.1 | 78.7 | 86.8 | 81.3 | 94.8 | 分隔解释 | 86.5 | 98.8 | 79.5 |
| 94.2 | 68.5 | 75.3 | 54.1 | 63.8 | 80.3 | 87.8 | 84.0 | 95.8 | 不打乱顺序 | 75.9 | 98.3 | 71.7 | 92.0 | 70.6 | 77.6 | 60.7 | 67.9 | 83.3 | 92.0 | 80.4 | 94.3 | | | | | | | | |

表 3：消融研究。

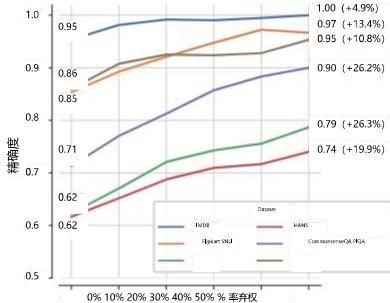


图 5：选择性预测在 T 检测分数上的准确率提升。

选择性预测的思路是放弃检测分数较低的LLM生成的答案，以保持剩余实例的更高准确性。如图 5 所示，我们展示了通过放弃 T 中检测分数最低的 0% - 50% 实例来维持剩余实例的准确性。我们可以观察到，随着放弃实例百分比的提高，这些数据集的准确性逐渐提高约 10% - 30%，IMDB 甚至达到了 100% 的准确性。自然地，对于准确性较低的数据集，提高的难度通常小于准确性较高的数据集。结果表明，T 在选择性预测场景中具有强大的应用潜力。

对 T 的鲁棒性分析。我们从三个方面评估 T 的鲁棒性：不同的目标答案、不同的LLMs以及参数敏感性。此外，我们在附录 C 中考察了提示的敏感性。

首先，目标答案 a 的生成在LLM随机性下可能有所不同，例如将温度设置为大于 0。我们通过利用不同的目标答案来验证 T 的鲁棒性，即 Self-cons

(a) 和 CoT-cons (a) 的多数答案，如表 4 所示。我们可以观察到以下情况。1) 对于这两组目标答案，T 在大多数情况下都优于基线，显示出其有效性。2) 不同的目标答案可能具有非常不同的自我检测性能。具体来说，与其它目标答案相比，CommonsenseQA 上的 T 和 CoT-cons 的 AUROC 值有显著下降，这可能是由于与 CoT 解释的多数投票有关。

| Flipkart HANS 常识问答 | |
|--------------------|----------|
| 自治 | 72.7 |
| 80.4 | Top-K 动词 |
| a | 69.5 |
| 主题一致性 | |
| 79.2 | 57.8 |
| a | 84.3 |
| 74.3 Top-K 动词 | |
| 83.9 | 53.3 |
| 67.5 | |
| T | |
| 84.3 | 69.2 |
| 75.0 | |

表 4：针对两个不同目标答案的 AUROC。

| Flipkart | HANS | CommonsenseQA | AUROC↑ | PRAUC↑ | AUROC↑ | PRAUC↑ | AUROC↑ | PRAUC↑ |
|-----------|------|---------------|--------|--------|--------|--------|--------|--------|
| CoT Cons. | 73.4 | 88.8 | 66.4 | 87.5 | 83.1 | 97.0 | 81.1 | 92.1 |

| | | | | | | | | | | | | | | | | | |
|------|------|------|------------|------|------|------|------|------|------|------|------|------|------|---|------|------|------|
| 混合 | 80.4 | 92.0 | 69.9 | 89.4 | 79.4 | 97.2 | CAPE | 82.3 | 92.7 | 82.4 | 94.0 | 80.0 | 96.8 | T | 83.3 | 93.4 | 82.0 |
| 93.9 | 72.5 | 96.0 | + Top-K 动词 | 82.7 | 93.2 | 80.9 | 93.9 | 81.0 | 97.6 | | | | | | | | |

表 5：Flipkart、HANS 和 CommonsenseQA 在 GLM-4 上的性能比较。

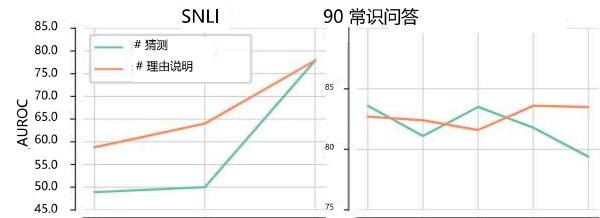


图 6：参数敏感性，即改变 p 中的论证数量和猜测数量。

原文中重复的 “the” 在翻译时可以省略，因为中文不会出现重复的定冠词。以下是翻译：

其次，我们在不同的LLMs上评估 T。

表 5 展示了 Flipkart、HANS 和 CommonsenseQA 在 GLM-4 上的性能比较。我们可以观察到，在LLMs的不同组合中，将 T 与 PE 或 Top-K Verb 结合使用，相较于其他方法表现更优，验证了其有效性。此外，自我检测能力在 LLMs上可能存在很大差异，例如，T 在 GLM-4 上对 HANS 的 AUROC 值，相较于 GPT-3.5 上大幅提升。更多关于 Gemini 的结果可以在附录 D 中找到。

第三，我们通过改变 p 中论证的数量和猜测的数量来评估 T 的参数敏感性。我们在 CommonsenseQA (有五个答案选项) 和 SNLI (有三个答案选项) 上进行了实验。从图 6 中，我们可以观察到以下几点。1) 增加论证的数量可以提高两个数据集上的性能，表明足够的论证数量是必要的。

tions is vital for better self-detection. 2) Increasing the number of guesses results in a significant performance improvement on the SNLI dataset, revealing that enough number of guesses is demanded for the NLI task. 3) Comparably, the change in the number of guesses has a slight effect on the performance of the CommonsenseQA dataset, which is potentially because the CQA task is more objective than NLI.

6 Conclusion

In this paper, we tackled the over-trust issue of self-detection on black-box API LLMs. We categorized existing methods into two paradigms and pointed out their limitation of merely evaluating on LLM-generated answer with potential LLM over-trust. We proposed a novel paradigm to address this limitation by comprehensively evaluating the trustworthiness of multiple candidate answers in the answer space. Following our paradigm, we presented a two-step framework T^3 by asking LLM to reflect and justify the validity of each answer for joint confidence calibration. Our framework achieved improved self-detection performance over compared methods and was combined with existing methods for further improvement. In future work, we will explore the combination of T^3 with more methods, and its utility in white-box LLMs.

指出足够的论证数量对于更好的自我检测至关重要。2) 在 SNLI 数据集上, 增加猜测次数显著提高了性能, 表明 NLI 任务需要足够的猜测次数。3) 相比之下, 猜测次数的变化对 CommonsenseQA 数据集的性能影响较小, 这可能是由于 CQA 任务比 NLI 任务更客观。

第六章 结论

在本文中, 我们解决了黑盒 API LLMs 自检测中的过度信任问题。我们将现有方法分为两种范式, 并指出了它们仅通过在 LLM 生成的答案上进行评估, 存在潜在 LLM 过度信任的局限性。我们提出了一种新颖的范式来解决这个问题, 通过全面评估答案空间中多个候选答案的可信度。遵循我们的范式, 我们提出了一个两步框架 T, 通过要求 LLM 反思和论证每个答案的有效性来进行联合置信度校准。我们的框架在自我检测性能上优于比较方法, 并与现有方法相结合以进一步改进。在未来工作中, 我们将探索 T 与更多方法的结合, 以及其在白盒 LLMs 中的实用性。

Limitations

Our work has several limitations. Firstly, our research scope is limited to the self-detection for black-box API LLM. While our framework is suitable for many state-of-the-art LLMs in this form, it might not be optimal for white-box LLMs, which offer access to more model information, thus limiting its broader applicability. Secondly, the utility of self-detection is not primarily studies in this work. Although we demonstrate the utility of detection scores in selective prediction scenarios, the challenge still lies in leveraging them to enhance task accuracy or enable LLM self-correction, calling for further exploration. Lastly, our framework lacks consideration in prompt optimization for self-detection, an area where future self-detection methods are expected to consider.

Ethics Statement

Our ethical concerns involve the following. First, our experimental results are mainly obtained in English datasets, where the applicability on other languages are not comprehensively evaluated. Secondly, our research scope is black-box API LLMs, where open-sourced LLMs are more advocated for its reproducibility. Finally, the self-detection of LLM may mislead people to blindly trust LLM and easily accept untrustable answers, causing potential harms.

Acknowledgements

This work is supported by the NExT Research Center. We thank the reviewers for their constructive feedback.

References

- Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyam Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.
- Ayush Agrawal, Lester Mackey, and Adam Tauman Kalai. 2023. Do language models know when they’re hallucinating references? *arXiv preprint arXiv:2305.18248*.
- Shengnan An, Zexiong Ma, Zeqi Lin, Nanning Zheng, Jian-Guang Lou, and Weizhu Chen. 2023. Learning from mistakes makes llm better reasoner. *arXiv preprint arXiv:2310.20689*.
- Nishant Balepur, Shramay Palta, and Rachel Rudinger. 2023. It’s not easy being wrong: Evaluating process of elimination reasoning in large language models. *arXiv preprint arXiv:2311.07532*.
- Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wen-liang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, et al. 2023. A multi-task, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. *arXiv preprint arXiv:2302.04023*.
- Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. 2020. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439.
- Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.
- Kendrick Boyd, Kevin H Eng, and C David Page. 2013. Area under the precision-recall curve: point estimates and confidence intervals. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2013, Prague, Czech Republic, September 23-27, 2013, Proceedings, Part III 13*, pages 451–466. Springer.
- Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.
- Jiuhai Chen and Jonas Mueller. 2023. Quantifying uncertainty in answers from any language model via intrinsic and extrinsic confidence assessment. *arXiv preprint arXiv:2308.16175*.
- Soham Dan and Dan Roth. 2021. On the effects of transformer size on in-and out-of-domain calibration. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2096–2101.
- Shrey Desai and Greg Durrett. 2020. Calibration of pre-trained transformers. *arXiv preprint arXiv:2003.07892*.
- Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhang Qiu, Zhilin Yang, and Jie Tang. 2022. Glm: General language model pretraining with autoregressive blank infilling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 320–335.
- Jinhao Duan, Hao Cheng, Shiqi Wang, Chenan Wang, Alex Zavalny, Renjing Xu, Bhavya Kailkhura, and Kaidi Xu. 2023. Shifting attention to relevance: Towards the uncertainty estimation of large language models. *arXiv preprint arXiv:2307.01379*.

限制

我们的工作存在一些局限性。首先，我们的研究范围仅限于对黑盒 API LLM 的自我检测。虽然我们的框架适用于许多在此形式下处于领先水平的 LLMs，但它可能不适合白盒 LLMs，因为白盒 API 提供了更多模型信息，这限制了其更广泛的应用。其次，自我检测的实用性在本工作中并未作为主要研究内容。尽管我们展示了检测分数在选择性预测场景中的效用，但如何利用这些分数来提高任务准确性或实现 LLM 自我校正仍然是一个挑战，需要进一步探索。最后，我们的框架在自我检测的提示优化方面缺乏考虑，这是未来自我检测方法预期会考虑的一个领域。

伦理声明

我们的伦理担忧包括以下几点。首先，我们的实验结果主要是在英语数据集上获得的，其他语言的适用性尚未进行全面评估。其次，我们的研究范围是黑盒 API LLMs，而开源的 LLMs 因其可重复性而更受推崇。最后，LLM 的自我检测可能会误导人们盲目信任 LLM，并轻易接受不可信的答案，造成潜在的危害。

致谢

这项工作得到了 NExT 研究中心的支持。感谢审稿人提出的建设性意见。

参考文献

Josh Achiam、Steven Adler、Sandhini Agarwal、Lama Ahmad、Ilge Akkaya、Florencia Leoni Aleman、Diogo Almeida、Janko Altenschmidt、Sam Altman、Shyamal Anadkat 等人 2023 年。Gpt-4 技术 arXiv 预印本 arXiv:2303.08774。

阿尤什·阿格拉瓦尔、莱斯特·麦凯和亚当·塔乌曼·卡拉伊。2023 年。语言模型知道它们在产生幻觉引用时吗？arXiv 预印本 arXiv:2305.18248。

安圣楠、马泽雄、林泽琪、郑南宁、楼建光、陈伟竹。2023。从错误中学习使 lIIm 成为更好的推理者。arXiv 预印本 arXiv:2310.20689。

尼尚特·巴勒普、沙拉姆亚·帕尔塔和雷切尔·鲁丁格。

2023 年：犯错不易：评估大规模语言模型中的排除法推理过程。
arXiv 预印本 arXiv:2311.07532。

叶金邦、塞缪尔·卡亚维加、李奈英、戴文良、苏丹、布莱恩·威尔、洛维尼娅·霍利、季子威、余铁正、钟伟利、等。2023。ChatGPT 在推理、幻觉和交互方面的多任务、多语言、多模态评估。arXiv 预印本 arXiv:2302.04023。

杨纳坦·比斯克、罗万·泽勒、高建峰、崔叶金、等人。2020 年。Piqa：在自然语言中推理物理常识。在《会议论文集》中。

AAAI 人工智能会议，第 34 卷，第 7432-7439 页。

鲍曼, S.R., 安格利, 加博尔, 波茨, 克里斯托弗, 和曼宁, 克里斯托弗.D. 2015. 用于学习自然语言推理的大型标注语料库。

在 2015 年实证自然语言处理会议论文集中，第 632-642 页，葡萄牙，里斯本。计算语言学协会。

凯文·H·恩格、凯文德·博伊德、C·大卫·佩奇。2013 年。面积下方的精确率-召回率曲线：点估计和置信区间。在《机器学习与数据库知识发现：欧洲会议论文集》中：

《第 13 部分：2013 年 ECML PKDD 会议论文集，2013 年 9 月 23 日至 27 日，捷克共和国布拉格，Springer 出版社，第 451-466 页。》

汤姆·B·布朗、本杰明·曼、尼克·赖特、梅拉妮·苏比亚、贾里德·卡普兰、普拉夫拉·达里瓦尔、阿温德·尼尔卡坦、普拉纳夫·希亚姆、吉里什·萨斯特里、阿曼达·阿斯凯尔、等人。2020 年。语言模型是零样本学习者。arXiv 预印本 arXiv:2005.14165。

九海陈和约纳斯·穆勒。2023 年。通过内在和外在置信度评估量化任何语言模型答案的不确定性。arXiv 预印本 arXiv:2308.16175。

索汉·丹和丹·罗斯。2021 年。关于 Transformer 大小对域内和域外校准的影响。

《计算语言学协会发现：EMNLP 2021 论文集，第 2096-2101 页。》

Shrey Desai 和 Greg Durrett. 2020. 预训练变换器的校准。arXiv 预印本 arXiv:2003.07892.

郑晓杜、钱宇杰、刘晓、丁明，

乔杰中、杨志林、唐杰. 2022. Glm：基于自回归空白填充的通用语言模型预训练. 在《计算语言学协会第 60 届年会论文集》（第 1 卷：长篇论文），第 320-335 页。

段金豪、程浩、王诗琪、王晨安、亚历克斯·扎瓦利尼、徐仁静、巴维娅·凯尔卡尔、徐凯迪。2023。聚焦相关性：迈向大型语言模型的不确定性估计。arXiv 预印本 arXiv:2307.01379。

- Shangbin Feng, Weijia Shi, Yike Wang, Wenzuan Ding, Vidhisha Balachandran, and Yulia Tsvetkov. 2024. Don’t hallucinate, abstain: Identifying llm knowledge gaps via multi-lm collaboration. *arXiv preprint arXiv:2402.00367*.
- Xiang Gao, Jiaxin Zhang, Lalla Mouatadid, and Kamalika Das. 2024. Spuq: Perturbation-based uncertainty quantification for large language models. *arXiv preprint arXiv:2403.02509*.
- Olga Golovneva, Moya Peng Chen, Spencer Poff, Martin Corredor, Luke Zettlemoyer, Maryam Fazel-Zarandi, and Asli Celikyilmaz. 2022. Roscoe: A suite of metrics for scoring step-by-step reasoning. In *The Eleventh International Conference on Learning Representations*.
- Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. 2017. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR.
- Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In *International Conference on Learning Representations*.
- Bairu Hou, Yujian Liu, Kaizhi Qian, Jacob Andreas, Shiyu Chang, and Yang Zhang. 2023. Decomposing uncertainty for large language models through input clarification ensembling. *arXiv preprint arXiv:2311.08718*.
- Mengting Hu, Zhen Zhang, Shiwan Zhao, Minlie Huang, and Bingzhe Wu. 2023. Uncertainty in natural language processing: Sources, quantification, and applications. *arXiv preprint arXiv:2306.04459*.
- Jiaxin Huang, Shixiang Gu, Le Hou, Yuexin Wu, Xuezhi Wang, Hongkun Yu, and Jiawei Han. 2023a. Large language models can self-improve. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1051–1068, Singapore. Association for Computational Linguistics.
- Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. 2023b. Large language models cannot self-correct reasoning yet. *arXiv preprint arXiv:2310.01798*.
- Yuheng Huang, Jiayang Song, Zhijie Wang, Huaming Chen, and Lei Ma. 2023c. Look before you leap: An exploratory study of uncertainty measurement for large language models. *arXiv preprint arXiv:2307.10236*.
- Mingjian Jiang, Yangjun Ruan, Sicong Huang, Saifei Liao, Silvia Pitis, Roger Baker Grosse, and Jimmy Ba. 2023. Calibrating language models via augmented prompt ensembles.
- Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. 2021. How can we know when language models know? on the calibration of language models for question answering. *Transactions of the Association for Computational Linguistics*, 9:962–977.
- Jaehun Jung, Lianhui Qin, Sean Welleck, Faeze Brahman, Chandra Bhagavatula, Ronan Le Bras, and Yejin Choi. 2022. Maieutic prompting: Logically consistent reasoning with recursive explanations. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1266–1279.
- Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schieber, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*.
- Divyansh Kaushik, Eduard Hovy, and Zachary Lipton. 2019. Learning the difference that makes a difference with counterfactually-augmented data. In *International Conference on Learning Representations*.
- Muhammad Khalifa, Lajanugen Logeswaran, Moon-tae Lee, Honglak Lee, and Lu Wang. 2023. Grace: Discriminator-guided chain-of-thought reasoning.
- Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *The Eleventh International Conference on Learning Representations*.
- Junyi Li, Xiaoxue Cheng, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2023a. Halueval: A large-scale hallucination evaluation benchmark for large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6449–6464.
- Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. 2024a. Inference-time intervention: Eliciting truthful answers from a language model. *Advances in Neural Information Processing Systems*, 36.
- Loka Li, Guangyi Chen, Yusheng Su, Zhenhao Chen, Yixuan Zhang, Eric Xing, and Kun Zhang. 2024b. Confidence matters: Revisiting intrinsic self-correction capabilities of large language models. *arXiv preprint arXiv:2402.12563*.
- Moxin Li, Wenjie Wang, Fuli Feng, Hanwang Zhang, Qifan Wang, and Tat-Seng Chua. 2023b. Hypothetical training for robust machine reading comprehension of tabular context. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1220–1236, Toronto, Canada. Association for Computational Linguistics.
- Yifei Li, Zeqi Lin, Shizhuo Zhang, Qiang Fu, Bei Chen, Jian-Guang Lou, and Weizhu Chen. 2023c. Making language models better reasoners with step-aware verifier. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5315–5333.

- 商斌风, 石伟佳, 王毅科, 丁文轩, 维迪沙·巴拉查兰, 尤莉亚·采特科夫。2024 年。
不要幻想, 戒除: 通过多渠道协作识别 ILM 知识差距。
arXiv 预印本 arXiv:2402.00367。
- 向高, 张嘉欣, 拉拉·穆阿塔迪德, 卡玛利卡·达斯。2024。
Spuq: 基于抗动的针对大型语言模型的不确定性量化。
arXiv 预印本 arXiv:2403.02509。
- 奥莉加·戈洛夫涅娃、彭晨、斯宾塞·波夫、马丁·科雷多、
卢克·泽特莱莫耶、玛丽亚姆·法泽尔扎尔迪、阿斯利·切利
吉尔马兹。2022 年。莫斯科: 一个
一套用于评估逐步推理的指标。
第 11 届国际学习表示会议
- 段国, 杰夫·普利斯, 孙宇, 基利安·Q·魏因伯格。2017。
《关于现代神经网络的校准》
在国际机器学习大会上发表, 第 1321-1330 页。
PMLR。
- Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, 和
Yejin Choi. 2020. 神经文本的奇异案例——
在国际学习表示会议上。
- 巴鲁·侯, 刘宇健, 钱凯志, 雅各布·安德烈亚斯, 常诗
雨, 张洋。2023. 通过输入澄清集成分解大型语言模型
的不确定性。arXiv 预印本 arXiv:2311.08718。
- 胡梦婷, 张振, 赵世旺, 黄敏丽, 吴冰哲。2023. 自然
语言处理中的不确定性: 来源、量化与应用。arXiv 预
印本 arXiv:2306.04459。
- 黄家新, 顾诗翔, 侯立, 吴越新, 王雪志, 余红坤, 韩
家炜。2023a. 大型语言模型可以自我改进. 在《会议论
文集》
- 2023 年实证自然语言处理会议论文集, 第 1051-
黄杰, 陈新云, 米沙罗普·米什拉, 郑怀秀·史蒂文, 余
伟, 宋新英, 周登尼。2023b. 《大型语言模型尚不能
自我纠正推理》。arXiv 预印本 arXiv:2310.01798。
- 黄宇恒, 宋佳洋, 王志杰, 陈华明, 马磊。2023c. 跳跃
之前先看看: 大型语言模型不确定性的探索性研
究。arXiv 预印本 arXiv:2307.10236。
- 江明健, 阮阳君, 黄思聪, 廖赛飞, 皮特斯·西尔维乌,
罗杰·贝克·格罗斯, 以及巴吉米·巴。2023. 通过增强提
示集校准语言模型。
- 郑宝江、远木耕、丁海波、内布格·格雷厄姆。2021。
《我们如何知道语言模型是否知晓? ——关于语言模型
校准》
- 关于问答。计算语言学协会学报, 第 9 卷, 第 962-
977 页。
- 姜焕俊, 秦连辉, 肖恩·韦勒克, 法泽·布拉曼, 钱德拉·巴
格瓦图拉, 罗南·勒布拉斯, 以及叶金·崔。2022 年。助产
式提示: 具有递归解释的逻辑一致推理。
- 《2022 年实证自然语言处理会议论文集, 第 1266-
1279 页》
- Saurav Kadavath, Tom Conerly, Amanda Askell,
Tom Henighan, Dawn Drain, Ethan Perez,
Nicholas Schiefer, Zac Hatfield-Dodds, Nova
DasSarma, Eli Tran-Johnson 等人 2022. 语言模型 (主
要) 知道它们所知道的内容。arXiv 预印本
arXiv:2207.05221。
- """
卡乌纳克·纳维拉·反重力·摩根·数据科学·班·拉·差异。
""在《国际学习表示会议》上发表。
- 穆罕默德·卡利法, 拉贾努根·洛格沙兰, 月亮-
李泰, 李洪·Iak, 王璐。2023. Grace: 判别器引导的链
式思维推理。
- 洛伦茨·库恩、亚里安·加尔和塞巴斯蒂安·法尔夸尔。2023 年。
语义不确定性: 自然语言生成中的语言不变性用于不确
定性估计。
第 11 届国际学习表示会议
- 李俊毅, 程晓雪, 赵新伟, 聂建云, 文继荣。2023a.
Halueval: 大规模幻觉评估基准
在 2023 年实证自然语言处理会议论文集中, 第
6449-6464 页。
- 李肯尼、奥姆·帕特尔、费尔南达·维加斯、汉斯彼得·皮斯
特、马丁·瓦滕伯格。2024a. 推断时间干预: 从人工智能
中获取真实回答的方法
《神经信息处理系统中的语言模型。神经信息处理系
统会议论文集, 第 36 卷。》
- 李洛卡, 陈广义, 苏玉生, 陈振豪, 张一璇, 邢志宏, 张
坤。
2024b. 信心很重要: 重新审视大型语言模型的内在自
我校正能力
arXiv 预印本 arXiv:2402.12563。
- 李墨轩, 王文杰, 冯福丽, 张汉旺, 王奇凡, 以及蔡天
森。2023b. 表格上下文鲁棒性机器阅读理解的假设训
练。在《协会发现》中。
- 计算语言学会议: ACL 2023, 第几页
1220-1236 年, 加拿大, 多伦多。计算语言学协会。
- 李一飞, 林泽琪, 张世卓, 傅强, 陈贝, 楼建光, 陈伟
竹。2023c. 通过步骤感知提升语言模型的推理能力
在《计算语言学协会第 61 届年度会议论文集》(第 1
卷: 长篇论文) 第 5315-5333 页。

- Stephanie Lin, Jacob Hilton, and Owain Evans. 2022a. Teaching models to express their uncertainty in words. *Transactions on Machine Learning Research*.
- Stephanie Lin, Jacob Hilton, and Owain Evans. 2022b. Truthfulqa: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252.
- Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. 2023. Generating with confidence: Uncertainty quantification for black-box large language models. *arXiv preprint arXiv:2305.19187*.
- Zhan Ling, Yunhao Fang, Xuanlin Li, Zhiao Huang, Mingu Lee, Roland Memisevic, and Hao Su. 2023. **Deductive verification of chain-of-thought reasoning**. In *Thirty-seventh Conference on Neural Information Processing Systems*.
- Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. **Learning word vectors for sentiment analysis**. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.
- Potsawee Manakul, Adian Liusie, and Mark Gales. 2023. **SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models**. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9004–9017, Singapore. Association for Computational Linguistics.
- Christopher Manning and Hinrich Schütze. 1999. *Foundations of statistical natural language processing*. MIT press.
- Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. **Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference**. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy. Association for Computational Linguistics.
- Ning Miao, Yee Whye Teh, and Tom Rainforth. 2023. Selfcheck: Using llms to zero-shot check their own step-by-step reasoning. *arXiv preprint arXiv:2308.00436*.
- Julian Michael, Salsabila Mahdi, David Rein, Jackson Petty, Julien Dirani, Vishakh Padmakumar, and Samuel R Bowman. 2023. Debate helps supervise unreliable experts. *arXiv preprint arXiv:2311.08702*.
- Sabrina J Mielke, Arthur Szlam, Emily Dinan, and Y-Lan Boureau. 2022. Reducing conversational agents’ overconfidence through linguistic calibration. *Transactions of the Association for Computational Linguistics*, 10:857–872.
- Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. **FActScore: Fine-grained atomic evaluation of factual precision in long form text generation**. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12076–12100, Singapore. Association for Computational Linguistics.
- Ansong Ni, Srini Iyer, Dragomir Radev, Veselin Stoyanov, Wen-tau Yih, Sida Wang, and Xi Victoria Lin. 2023. Lever: Learning to verify language-to-code generation with execution. In *International Conference on Machine Learning*, pages 26106–26128. PMLR.
- OpenAI. 2024. **Chatgpt**.
- Liangming Pan, Michael Saxon, Wenda Xu, Deepak Nathani, Xinyi Wang, and William Yang Wang. 2023. **Automatically correcting large language models: Surveying the landscape of diverse self-correction strategies**. *CoRR*, abs/2308.03188.
- Jie Ren, Yao Zhao, Tu Vu, Peter J Liu, and Balaji Lakshminarayanan. 2023a. Self-evaluation improves selective generation in large language models. *arXiv preprint arXiv:2312.09300*.
- Ruiyang Ren, Yuhao Wang, Yingqi Qu, Wayne Xin Zhao, Jing Liu, Hao Tian, Hua Wu, Ji-Rong Wen, and Haifeng Wang. 2023b. Investigating the factual knowledge boundary of large language models with retrieval augmentation. *arXiv preprint arXiv:2307.11019*.
- Andy Shih, Dorsa Sadigh, and Stefano Ermon. 2023. Long horizon temperature scaling. *arXiv preprint arXiv:2302.03686*.
- Chenglei Si, Zhe Gan, Zhengyuan Yang, Shuohang Wang, Jianfeng Wang, Jordan Lee Boyd-Graber, and Lijuan Wang. 2022. Prompting gpt-3 to be reliable. In *The Eleventh International Conference on Learning Representations*.
- Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. Commonsenseqa: A question answering challenge targeting commonsense knowledge. 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 4149–4158.
- Katherine Tian, Eric Mitchell, Huaxiu Yao, Christopher Manning, and Chelsea Finn. 2023a. Fine-tuning language models for factuality. In *NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following*.
- Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D Manning. 2023b. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. *arXiv preprint arXiv:2305.14975*.

林斯蒂芬妮, 雅各布·希尔顿, 欧文·埃文斯。2022a。
教给模型表达其不确定性的方法
《词汇。机器学习研究交易》

林斯蒂芬妮, 雅各布·希尔顿, 欧文·埃文斯。2022b。
真实问答: 衡量模型如何模仿人类
在《计算语言学协会第 60 届年会论文集》(第 1
卷: 长篇论文) 第 3214-3252 页。

郑林, 舒布亨杜·特里维迪, 孙建明。2023 年。
《生成有信心: 黑盒大型语言模型的不确定性量
化》。arXiv 预印本 arXiv:2305.19187。

张凌, 方云浩, 李璇琳, 黄钊, 李明宇, 罗兰·梅米塞维
奇, 苏浩, 2023 年。
演绎验证思维链推理。
第三十七届神经信息处理系统大会

安德鲁·L·马萨斯, 雷蒙德·E·戴利, 彼得·T·范, 丹·黄, 安德
鲁·Y·吴, 克里斯托弗·波茨。
2011 年: 学习用于情感分析的词向量
《第 49 届计算语言学协会人机语言技术年会论文集:
第 142-150 页, 波特兰》

俄勒冈州, 美国。计算语言学协会。

Potsawee Manakul、Adian Liusie 和 Mark Gales. 2023
自检 GPT: 面向生成式大型语言模型的零资源黑盒幻
觉检测。
在 2023 年实证自然语言处理会议论文集中, 第
9004-9017 页, 新加坡。计算语言学协会。

Christopher Manning 和 Hinrich Schutze. 1999. 基础—
统计自然语言处理的基础。
麻省理工学院出版社

汤姆·麦科伊、埃莉·帕夫利克和塔尔·林岑。2019 年。因
错而正确: 诊断自然语言推理中的句法启发式。在《会议
论文集》中
第 57 届计算语言学协会年会, 第 3428-3448 页, 意
大利佛罗伦萨。计算语言学协会。

宁苗, 叶威仪, 汤姆·雷恩福斯。
2023 年, 自我检查: 使用llms进行零样本逐步推理检
查。arXiv 预印本 arXiv:2308.00436。

朱利安·迈克尔, 萨拉萨比拉·马迪, 大卫·雷因, 杰克逊·佩
蒂, 朱利安·迪拉尼, 维沙克·帕德马库马尔, 以及塞缪尔
·R·鲍曼。2023 年。辩论有助于监督
不可靠的专家。arXiv 预印本 arXiv:2311.08702。

米歇尔·J·米勒, 亚瑟·斯兹拉姆, 艾米丽·迪南, 以及杨兰·
布罗埃。2022 年。通过语言校准降低对话代理的过度自信。
协会计算语言学研究论文, 第 10 卷, 第 857-872
页。

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike
Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke
Zettlemoyer 和哈纳内·哈吉什尔齐。2023. FActScore: 长文
本生成中事实精确度的细粒度原子评估。载于 2023 年
实证自然语言处理会议论文集, 第 12076-12100 页,
新加坡。计算语言学协会。

安松·尼, 斯里尼·伊耶, 德拉戈米尔·拉代夫, 韦塞林·斯托
扬诺夫, 温-塔乌·叶, 王思达, 林西维多利亚。

2023 年, Lever: 通过执行验证语言到代码生成。国
际机器学习会议, 第 26106-26128 页。

PMLR.

OpenAI. 2024. Chatgpt.

潘亮明, 迈克尔·萨克斯, 许文达, 达卡·纳塔尼, 王欣
怡, 王阳。2023 年。

自动校正大型语言模型: 探讨多样化的自我校正策略概
览。arXiv, abs/2308.03188。

节仁, 赵耀, 武图, 刘彼得·刘, 和拉克希米纳拉扬纳安·
巴拉吉。2023a. 自我评估提升大型语言模型的选区生
成。arXiv 预印本 arXiv:2312.09300。

任瑞阳, 王宇豪, 曲英琪, 赵新宇, 刘静, 田浩, 吴
华, 温吉荣, 王海峰。2023b. 利用检索增强研究大型语
言模型的事实知识边界。arXiv 预印本
arXiv:2307.11019。

安迪·施, 多萨·萨迪格, 斯蒂法诺·埃尔蒙。2023 年。

长时程温度尺度。arXiv 预印本 arXiv:2302.03686。

程磊四, 张干, 杨正远, 王硕航, 王建锋, 乔丹·李·博伊
德-格雷伯, 和
王丽娟。2022. 引导 gpt-3 变得可靠。
第 11 届国际学习表示会议

阿隆·塔尔莫尔、乔纳森·赫齐格、尼古拉斯·卢里和乔纳森·
贝兰特。2019 年。《Commonsenseqa: 针对常识知识
的问答挑战》

在 2019 年北美计算语言学协会人机语言技术会议论文
集第 1 卷 (长篇和短篇论文) 第 4149-4158 页。

田凯瑟琳、埃里克·米切尔、姚华秀、克里斯托弗·曼宁和
切尔西·芬恩。2023a. 为事实性微调语言模型。在
NeurIPS 2023
工作坊: 指令调整与指令遵循。

田凯瑟琳、埃里克·米切尔、周安澜、阿奇特·夏尔马、拉
斐尔·拉菲洛夫、姚华秀、切尔西·芬恩和克里斯托弗·D·曼
宁。2023b. 只需请求校准: 从使用人类反馈微调的语言
模型中提取校准置信度分数的策略。arXiv 预印本
arXiv:2305.14975。

- Nirali Vaghani and Mansi Thummar. 2023. BFlipkart product reviews with sentiment dataset. <https://www.kaggle.com/dsv/4940809>.
- Danqing Wang and Lei Li. 2023. Learning from mistakes via cooperative study assistant for large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10667–10685.
- Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*.
- Yixuan Weng, Minjun Zhu, Fei Xia, Bin Li, Shizhu He, Shengping Liu, Bin Sun, Kang Liu, and Jun Zhao. 2023. Large language models are better reasoners with self-verification. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 2550–2575, Singapore. Association for Computational Linguistics.
- Yuxi Xie, Kenji Kawaguchi, Yiran Zhao, Xu Zhao, Min-Yen Kan, Junxian He, and Qizhe Xie. 2023. Self-evaluation guided beam search for reasoning. In *Thirty-seventh Conference on Neural Information Processing Systems*.
- Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2023. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. *arXiv preprint arXiv:2306.13063*.
- Polina Zablotskaia, Du Phan, Joshua Maynez, Shashi Narayan, Jie Ren, and Jeremiah Liu. 2023. On uncertainty calibration and selective generation in probabilistic neural summarization: A benchmark study. *arXiv preprint arXiv:2304.08653*.
- Hanlin Zhang, Yi-Fan Zhang, Yaodong Yu, Dhruv Madeka, Dean Foster, Eric Xing, Hima Lakkaraju, and Sham Kakade. 2023a. A study on the calibration of in-context learning. *arXiv preprint arXiv:2312.04021*.
- Jiaxin Zhang, Zhuohang Li, Kamalika Das, Bradley A. Malin, and Sricharan Kumar. 2023b. SAC\$^3\$: Reliable hallucination detection in black-box language models via semantic-aware cross-check consistency. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.
- Mozhi Zhang, Mianqiu Huang, Rundong Shi, Linsen Guo, Chong Peng, Peng Yan, Yaqian Zhou, and Xipeng Qiu. 2024. Calibrating the confidence of large language models by eliciting fidelity. *arXiv preprint arXiv:2404.02655*.
- Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, et al. 2023c. Siren’s song in the ai ocean: A survey on hallucination in large language models. *arXiv preprint arXiv:2309.01219*.
- Ruochen Zhao, Xingxuan Li, Shafiq Joty, Chengwei Qin, and Lidong Bing. 2023a. Verify-and-edit: A knowledge-enhanced chain-of-thought framework. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5823–5840, Toronto, Canada. Association for Computational Linguistics.
- Theodore Zhao, Mu Wei, J Samuel Preston, and Hoifung Poon. 2023b. Automatic calibration and error correction for large language models via pareto optimal self-supervision. *arXiv preprint arXiv:2306.16564*.
- Xinran Zhao, Hongming Zhang, Xiaoman Pan, Wenlin Yao, Dong Yu, Tongshuang Wu, and Jianshu Chen. 2024. Fact-and-reflection (far) improves confidence calibration of large language models. *arXiv preprint arXiv:2402.17124*.
- Yukun Zhao, Lingyong Yan, Weiwei Sun, Guoliang Xing, Chong Meng, Shuaiqiang Wang, Zhicong Cheng, Zhaochun Ren, and Dawei Yin. 2023c. Knowing what llms do not know: A simple yet effective self-detection method. *arXiv preprint arXiv:2310.17918*.
- Aojun Zhou, Ke Wang, Zimu Lu, Weikang Shi, Sichun Luo, Zipeng Qin, Shaoqing Lu, Anya Jia, Linqi Song, Mingjie Zhan, et al. 2023. Solving challenging math word problems using gpt-4 code interpreter with code-based self-verification. *arXiv preprint arXiv:2308.07921*.

A Details for compared methods.

A.1 LLM Hyperparameters.

For all LLMs, we set the maximum token as 200. For GPT-3.5 and Gemini, if sampling a single response ($N = 1$), we set the temperature as 0, and other hyperparameters as default. If sampling multiple responses, we sample $N = 30$ ($N = 5$ for Gemini due to API call limitation) responses with temperature as 1, which is only for Self-cons, CoT-cons, and P(True). Specially, for Self-detect we sample 15 rephrasing for each question with temperature as 1, and one answer for each rephrased question with temperature as 0, following the original paper. For GLM-4, if sampling a single response, we set the do_sample as False. If sampling a variety of responses, we set temperature as 0.9, top p as 0.9, and $N = 5$. Note that these LLM hyperparameters are not carefully tuned.

A.2 Dataset Detail.

Due to the cost limitation, we randomly sample 300 training data for each dataset in our experiments. For IMDB and SNLI datasets, we use the same

尼拉利·瓦加尼和曼西·图马尔。2023 年。BFlipkart 产品评论及情感数据集。

<https://www.kaggle.com/dsv/4940809>。

王丹清, 李磊. 2023. 通过协作学习助教从错误中学习

在 2023 年实证自然语言处理会议论文集中, 第 10667-10685 页。

王雪志, 魏杰, 舒尔曼斯, 吴国越, 蔡志坚, 纳兰, 乔安卡莎·乔杜里, 周登宇。2022 年。自洽性提升语言模型中的思维链推理。

第十一届国际学习表示会议

魏一璇, 朱敏军, 夏飞, 李斌, 何世柱, 刘胜平, 孙斌, 刘康, 赵军。

2023 年, 大型语言模型在自我验证方面的推理能力更强。在《协会发现》中.....

《计算语言学: EMNLP 2023, 第 2550-2575 页, 新加坡。计算语言学协会。》

谢宇熙, 川口健司, 赵一然, 赵旭, 甘明炎, 何俊贤, 谢奇哲。2023 年。

自我评估引导的推理光束搜索

第三十七届神经信息处理系统大会

苗雄、胡志远、陆新阳、李一飞、傅杰、何俊贤和黄 Bryan。2023 年。llms 能否表达其不确定性? llms 中置信度提取的实证评估。arXiv 预印本 arXiv:2306.13063。

Polina Zablotskaia, Du Phan, Joshua Maynez, Shashi Narayan, Jie Ren, 以及 Jeremiah Liu. 2023. 关于概率神经摘要中的不确定性校准和选择性生成: 一项基准研究。

arXiv 预印本 arXiv:2304.08653。

张汉林, 张毅凡, 余耀东, Dhruv Madeka, 福斯特, Xing Eric, Lakkaraju Hima, Kakade Sham。2023a. 《情境学习的校准研究》。arXiv 预印本 arXiv:2312.04021。

张嘉欣, 李卓航, 达卡玛利卡, 布拉德利·A

马林, 和库马尔斯里查兰。2023b. SAC\$^3\$: 通过语义感知交叉一致性, 在黑盒语言模型中进行可靠的幻觉检测。

在 2023 年实证自然语言处理会议

张莫之, 黄面秋, 石润东, 郭林森, 彭冲, 严鹏, 周亚倩, 邱锡鹏。2024. 通过唤起保真度校准大型语言模型的置信度。arXiv 预印本 arXiv:2404.02655。

张越, 李亚夫, 崔磊阳, 蔡登, 刘乐茂, 傅廷晨, 黄新亭, 赵恩博, 张宇, 陈宇龙, 等。2023c. AI 海洋中的塞王之歌: 大型语言模型幻觉综述。arXiv 预印本 arXiv:2309.01219。

赵若晨, 李星轩, 沙菲克·乔蒂, 秦成伟, 丁立东。

2023a. 验证与编辑:

知识增强的思维链框架。

在《计算语言学协会第 61 届年度会议论文集》(第 1 卷: 长篇论文), 第 5823-5840 页, 加拿大多伦多。

计算语言学协会

Oscar Theodore, 魏 Mu, Preston J Samuel, Poon Hoifung。2023b. 通过帕累托最优自监督进行大型语言模型的自动校准和错误纠正。arXiv 预印本

arXiv:2306.16564。

赵欣然, 张红明, 潘晓曼, 姚文林, 余东, 吴同双, 陈建树。

2024 年, 事实与反思 (far) 提高了大型语言模型的置信度校准。arXiv 预印本 arXiv:2402.17124。

赵宇坤, 严凌勇, 孙伟伟, 邢国良, 孟崇, 王帅强, 支崇聪

程, 赵春, 尹大伟。2023c. 知道 llms 不知道什么: 一种简单而有效的自我检测方法。arXiv 预印本 arXiv:2310.17918。

周奥军, 王科, 陆子穆, 石伟康, 罗思春, 秦子鹏, 卢少庆, 贾安雅, 宋琳琪, 詹明杰, 等。2023. 利用 gpt-4 代码解释器及基于代码的自验证解决具有挑战性的数学文字问题。arXiv 预印本 arXiv:2308.07921。

详细比较方法的说明。

A.1 LLM 超参数。

对于所有 LLMs, 我们将最大 token 数设置为 200。

对于 GPT-3.5 和 Gemini, 如果采样单个响应

($N=1$), 我们将温度设置为 0, 其他超参数使用默认值。如果采样多个响应, 我们将采样 $N=30$ 个响应 (Gemini 由于 API 调用限制, 为 $N=5$), 温度设置为 1, 仅适用于 Self-cons、CoTcons 和 P(True)。特别地, 对于 Self-detect, 我们对每个问题采样 15 个重述, 温度设置为 1, 并对每个重述问题采样一个答案, 温度设置为 0, 遵循原始论文。对于 GLM-4, 如果采样单个响应, 我们将 do_sample 设置为 False。如果采样多种响应, 我们将温度设置为 0.9, top p 设置为 0.9, N=5。请注意, 这些 LLM 超参数并未经过精心调整。

A.2 数据集详细信息。

由于成本限制, 我们在实验中对每个数据集随机抽取了 300 个训练数据。对于 IMDB 和 SNLI 数据集, 我们使用相同的。

| | <i>N</i> | examples |
|---------------|----------|---|
| IMDB | 2 | positive negative |
| Flipkart | 2 | positive negative |
| SNLI | 3 | entailment, neutral, contradiction |
| HANS | 2 | entailment, non entailment |
| CommonsenseQA | 5 | (a) yard, (b) basement,
(c) kitchen, (d) living room, (e) garden |
| PIQA | 2 | (a) pour it onto a plate, (b) pour it into a jar |

Table 6: The number (*N*) and examples of candidate answers for each dataset.

randomly sampled 300 data sets as the CAD SA and NLI in the preliminary experiments. We will release the dataset splits. Table 6 shows the number and examples of candidate answers for each dataset.

A.3 Prompts

The basic instructions for different datasets are shown as below, where [] refers to specific task inputs.

- **IMDB:**

Given a piece of movie review, classify the attitude to the movie as Positive or Negative. [text]

- **Flipkart:**

Given a piece of text, classify the sentiment as Positive or Negative. [text]

- **SNLI:**

Determine whether the hypothesis is an entailment (can be logically inferred from the premise), a contradiction (cannot be true given the premise), or neutral (does not have enough information to determine its truth value). Premise: [premise] Hypothesis: [hypothesis].

- **HANS:**

Determine whether the second sentence in each pair logically follows from the first sentence. The output is either "entailment" if the second sentence logically follows from the first, or "not entailment" if it does not. Sentence 1: [sentence1]. Sentence 2: [sentence2].

- **CommonsenseQA:**

Read the given question and select the most appropriate answer by indicating the associated letter. Question: [question]. Answer choices: (a) a_1^q (b) a_2^q (c) a_3^q (d) a_4^q (e) a_5^q .

- **PIQA:**

Read the given question and select the most appropriate answer by indicating the associated letter. Question: [question]. Answer choices: (a) a_1^q (b) a_2^q .

The prompts for compared methods are shown below, where [instruction] denotes the task instruction with the task input, and [instruction_only] denotes the instruction without task input.

- **Self-cons:** [instruction].

- **CoT-cons:**

[instruction]. Please output strictly following this format: Explanation: [reasons for the sentiment label] Answer: [Positive or Negative]

- **Top-K Verb p^b :**

*The task is to [instruction_only]. Provide your n best guesses and the probability that each is correct (0.0 to 1.0) for the following question. Give ONLY the guesses and probabilities, no other words or explanation. For example:
G1: <first most likely guess, as short as possible; not a complete sentence, just the guess!>
P1: <the probability between 0.0 and 1.0 that G1 is correct, without any extra commentary whatsoever; just the probability!> ... GN: <N-th most likely guess, as short as possible; not a complete sentence, just the guess!>
PN: <the probability between 0.0 and 1.0 that GN is correct, without any extra commentary whatsoever; just the probability!> [question] [answer choices].*

- **P(True) p^t :**

The task is to [instruction]. Label: [label]. Is the label correct or incorrect?

- **Self-detect:**

For question rephrasing: Paraphrase the given sentence. Please make sure the paraphrased sentence has exactly the same meaning as the original sentence. [question]

For inference: [instruction].

- **CAPE:**

Provide your 2 best guesses and the probability that each is correct (0.0 to 1.0) for the following task. Give ONLY the guesses and probabilities, no other words or explanation. For example:

| N 个示例 |
|---|
| IMDB 2 正面 负面 Flipkart 2 正面 负面 SNLI 3 包含、中立、矛盾 HANS 2 包含、不包含 |
| (a) 5 码, (b) 地下室,
(c) 厨房, (d) 客厅, (e) 花园 PIQA 2 (a) 倒在
盘子里, (b) 倒进罐子里 |

表 6: 每个数据集的候选答案数量 (N) 及示例。

在初步实验中，我们随机抽取了 300 个数据集作为 CAD SA 和 NLI。我们将发布数据集的划分。表 6 显示了每个数据集中候选答案的数量和示例。

A.3 提示

以下展示了不同数据集的基本操作指南，其中[]表示特定任务的输入。

- IMDB:

给定一段电影评论，将对该电影的态度分类为正面或负面。[文本]

- Flipkart:

给一段文本，将其情感分类为正面或负面。[文本]

- SNLI:

确定假设是否是前提的蕴涵（可以从前提中逻辑地推断出来）、矛盾（在前提条件下不能为真）还是中立（没有足够的信息来确定其真值）。前提：[前提]，假设：[假设]。

- 汉斯:

判断每对句子中第二个句子是否逻辑上由第一个句子推导而来。输出结果为“蕴含”如果第二个句子逻辑上由第一个句子推导而来，或者“非蕴含”如果它不推导而来。句子 1: [sentence1]。句子 2: [sentence2]。

- 常识问答:

阅读以下问题，并选择最合适答案，通过指出相应的字母。问题：[问题]。选项：(a) a(b) a(c) a(d) a(e) a。

- PIQA:

阅读所给问题，并选择最合适答案，通过指出相应的字母。问题：[问题]。选项：(a) a(b) a.

以下是比較方法的提示，其中[指令]表示带有任务输入的任务指令，[指令 (无输入)]表示不带任务输入的指令。

- 自洽：[指令]。

- 源文本：

说明：[对情感标签原因的解释] 答案：[正面或负面]cons:
""

翻译：

- 顶K 动词 p:

任务为[属性仅限]。针对以下问题，提供你的最佳猜测及每个猜测正确的概率 (0.0 至 1.0)，仅此而已。例如：G1: <最有可能的猜测，尽可能简短；不是完整的句子，只是猜测！> P1: <g1> ... GN: <第 N 个最有可能的猜测，尽可能简短；不是完整的句子，只是猜测！> PN: [问题] [选项]。</g1>

- P(真) p:

任务是对[指令]进行仔细分析和深入理解。标签：[标签]。这个标签正确吗？

- 自检：

关于问题改写：请对所给句子进行释义。请确保释义后的句子与原句意义完全相同。[问题] 对于推理：[指令]。

- CAPE:

提供您对以下任务的 2 个最佳猜测，并给出每个猜测正确的概率 (0.0 至 1.0)。只给出猜测和概率，不要其他文字或解释。例如：
""

G1: <first most likely guess, as short as possible; not a complete sentence, just the guess!>

P1: <the probability between 0.0 and 1.0 that G1 is correct, without any extra commentary whatsoever; just the probability!> ... GN: <N-th most likely guess, as short as possible; not a complete sentence, just the guess!>

PN: <the probability between 0.0 and 1.0 that GN is correct, without any extra commentary whatsoever; just the probability!> Instruction: [instruction_only] [question]

- a_1^q (or A. a_1^q)

...

- a_N^q (or N. a_N^q)

Possible explanation 1: [e¹]

...

Possible explanation N: [e^N]

Correct Choice:

- $T^3 p^v$:

The task is to [instruction_only]. Provide your n best guesses and the probability that each is correct (0.0 to 1.0) for the following question. Give ONLY the guesses and probabilities, no other words or explanation. For example:

G1: <first most likely guess, as short as possible; not a complete sentence, just the guess!>

P1: <the probability between 0.0 and 1.0 that G1 is correct, without any extra commentary whatsoever; just the probability!> ... GN: <N-th most likely guess, as short as possible; not a complete sentence, just the guess!>

PN: <the probability between 0.0 and 1.0 that GN is correct, without any extra commentary whatsoever; just the probability!>

[question] [answer choices].

Possible explanation 1: [explanation 1].

...

Possible explanation N: [explanation N].

A.4 Additional Implementation Detail.

For T^3 and Top- K Verb, the N is set to the number of candidate answers for each dataset as in Table 6.

For the shuffling of the justification order in p^v , we use one original and one reversed order for T^3 on all datasets. For datasets with more than two justifications (SNLI and CommonsenseQA), we set the original justification order for SNLI as "entailment, neutral, contradiction" and follow the given answer choice order for CommonsenseQA

in the dataset.

CAPE is prompt ensemble for Top- K Verb. We follow the original paper to adopt two multi-choice template with alphabetic or itemized labels in addition to the original Top- K Verb prompt (See Section A.3). For each multi-choice template, we use the original and the reversed label orders. In total, the confidence score is an average of five prompts.

For $T^3 + PE$, we put T^3 into the multi-choice template with alphabetic labels, and use two reversed label orders and 2 reversed justification orders, in total four prompts.

The number of API calls for different methods are shown in Table 7. We can observe that compared with other methods T^3 does not incur large increase in number of calls. In our experiments, the maximum value of N is 5. Considering its effectiveness, the cost of T^3 is reasonable.

| | Self-cons | CoT-cons | Top- K Verb | P(True) | Hybrid |
|-------------|-----------|----------|---------------|-----------------------------------|------------|
| # call | 30 | 30 | 1 | 30 | 31 |
| Self-detect | | CAPE | T^3 | $T^3 + \text{Top-}K \text{ Verb}$ | $T^3 + PE$ |
| # call | 30 | 5 | N+2 | N+3 | N+4 |

Table 7: Comparison on the number of API calls of compared methods, where N denotes the number of choices for different datasets.

B Implementation Detail for Preliminary Experiments.

For the preliminary experiments, we randomly sample 300 instances from the training set of CAD SA and NLI, respectively. For those original questions with more than one counterfactual questions, we randomly select one counterfactual question for experiment. The prompts can be viewed in Section A.3. CAD SA is annotated from IMDB, and CAD NLI is annotated from SNLI. The w/ cf is based on Top- K Verb, which is better calibrated than Self-cons. For w/ cf, we obtain the Top- K Verb outputs for counterfactual and original questions, respectively. We use the guess with the largest probability in the response as the answer to \bar{q} , and the probability as its confidence score. The LLM is GPT-3.5 (gpt-3.5-1106). See Section A.1 for LLM hyperparameters.

| | PIQA | HANS | Flipkart |
|-------|----------------|----------------|----------------|
| p^e | 84.2 ± 2.0 | 62.7 ± 4.3 | 78.0 ± 2.2 |
| p^v | 83.0 ± 0.5 | 68.3 ± 1.7 | 81.2 ± 0.3 |

Table 8: The average and standard deviation of AUROC for T^3 with different rephrasing of prompts on GPT-3.5.

G1: <最有可能的初步猜测，尽可能简短；不是完整的句子，只是猜测！>

P1: <g1> ... GN: <第 N 个最可能的猜测，尽可能简短；不是完整的句子，仅仅是猜测！>
PN: 指令：[仅指令] [问题]</g1>

- a (或 A. a)

...

- a(或 N. a) 可能的解释 1: [e]

... 可能的解释 N: [e] 正确选项：

- T p:

任务为[指令仅限]。针对以下问题，提供你的最佳猜测及每个猜测正确的概率（0.0 至 1.0），仅此而已。例如：G1: <最有可能的猜测，尽可能简短；不是完整的句子，只是猜测！> P1: <g1> ... GN: <第 N 个最有可能的猜测，尽可能简短；不是完整的句子，只是猜测！> PN: [问题] [选项]。</g1>

可能的解释 1: [解释 1]。

...

可能的解释 N: [解释 N]。

A.4 补充实施细节。

对于 T 和 Top-K 动词，N 被设置为每个数据集的候选答案数量，如表 6 所示。

对于 p 中论证顺序的打乱，我们在所有数据集上使用一个原始顺序和一个反转顺序的 T。对于具有超过两个论证的数据集（SNLI 和 CommonsenseQA），我们将 SNLI 的原始论证顺序设置为“蕴涵、中立、矛盾”，并遵循 CommonsenseQA 的给定答案选项顺序。

在数据集中。

CAPE 是针对 Top-K 动词的提示集成。我们遵循原始论文，除了原始的 Top-K 动词提示外，还采用了两种带有字母或列举标签的多选题模板（参见 A.3 节）。对于每个多选题模板，我们使用原始标签顺序和反转标签顺序。总而言之，置信度分数是五个提示的平均值。

对于 T + PE，我们将 T 放入带有字母标签的多选题模板中，并使用两种颠倒的标签顺序和两种颠倒的对齐顺序，总共四个提示。

表 7 展示了不同方法的 API 调用次数。我们可以观察到，与其他方法相比，T 方法并未引起调用次数的大幅增加。在我们的实验中，N 的最大值为 5。考虑到其有效性，T 方法的成本是合理的。

自洽 CoT-一致 Top-K 动词 P(真) 混合

拨打 30 30 1 30 31

自我检测 CAPE T T + Top-K 动词 T + PE # 调用 30-5-N+2 N+3 N+4

表 7：比较不同方法在 API 调用次数上的差异，其中 N 表示不同数据集的选择数量。

“初步实验的实施细节。”

对于初步实验，我们从 CAD SA 和 NLI 的训练集中分别随机抽取了 300 个实例。对于那些具有多个反事实问题的原始问题，我们随机选择一个反事实问题进行实验。提示内容可在 A.3 节查看。CAD SA 来自 IMDB 的标注，CAD NLI 来自 SNLI 的标注。w/ cf 基于 Top-K Verb，其校准效果优于 Self-cons。对于 w/ cf，我们分别获取反事实问题和原始问题的 Top-K Verb 输出。我们以响应中概率最大的猜测作为答案。

\bar{q} ，其置信度分数为概率。LLM 是 GPT-3.5 (gpt-3.5-1106)。见 A.1 节了解 LLM 的超参数。

PIQA HANS Flipkart p84.2 ± 2.0 62.7 ± 4.3
-78.0 ± 2.2 p83.0 ± 0.5 68.3 ± 1.7 81.2 ±
0.3

表 8：在 GPT-3.5 上对不同提示重述的平均 AUROC 及其标准差。

C Prompt Sensitivity

We examine the prompt sensitivity of p^e and p^v by rephrasing each of them three times with ChatGPT⁶ and compute the average and standard deviation of AUROC, as shown in Table 8. We can observe the followings. 1) The variation of prompts has a mild effect on the performance of T^3 . Across the three datasets, HANS is the most sensitive to prompt rephrasing, potentially related to its lower AUROC performance. 2) The change of p^e has larger impact on the detection performance than p^v . This is probably because the justifications generated by p^e have a larger space of variation than the outputs of p^v , *i.e.*, guesses and probabilities.

| | Flipkart | | PIQA | | CommonsenseQA | |
|--------------|----------|-------|-------|-------|---------------|-------|
| | AUROC | PRAUC | AUROC | PRAUC | AUROC | PRAUC |
| CoT-cons | 67.6 | 89.1 | 85.8 | 96.4 | 72.2 | 86.0 |
| Top-K Verb | 84.0 | 94.9 | 64.7 | 91.8 | 67.0 | 83.9 |
| Hybrid | 85.0 | 95.5 | 77.0 | 96.1 | 74.5 | 87.9 |
| CAPE | 81.5 | 94.5 | 86.5 | 97.8 | 77.3 | 89.5 |
| T^3 | 82.0 | 93.9 | 82.6 | 96.5 | 73.4 | 89.3 |
| + Top-K Verb | 83.1 | 94.6 | 77.2 | 96.1 | 76.3 | 91.2 |
| + CAPE | 74.8 | 92.1 | 90.5 | 98.3 | 77.5 | 91.1 |

Table 9: Performance comparison of Gemini on Flipkart, PIQA and CommonsenseQA.

D Additional Results on Different LLMs

In addition to GPT-3.5 and GLM-4, we show the results of Gemini on three datasets. From Table 9, we can observe that although T^3 outperforms all compared methods in PIQA and CommonsenseQA, it does not outperform all compared methods on Flipkart. By analyzing the outputs, we discover that Gemini cannot always follow the instruction to perform reflection and generated justification for the designated answer. Instead, it tends to perform answer prediction and followed by an explanation on its predicted answer. Without effective reflection and justification from different answers’ perspectives, the effectiveness of T^3 is diminished. Therefore, the effectiveness of T^3 depends on the ability of the specific LLM in following the instructions in Table 1.

E Additional Baselines.

We present the results of more baselines on GPT-3.5, including the white-box average token probability (Manakul et al., 2023), and the black-box induced consistency and self-probing (Xiong et al., 2023). From Table 10, we can observe that

the white-box method largely underperforms the black-box baselines across three tasks, indicating its limited capability on large LLMs with strong instruction-following abilities.

| | Flipkart | | HANS | | CommonsenseQA | |
|--------------|----------|-------|-------|-------|---------------|-------|
| | AUROC | PRAUC | AUROC | PRAUC | AUROC | PRAUC |
| Avg Tok Prob | 32.4 | 78.0 | 56.0 | 65.5 | 43.3 | 67.3 |
| Induced-cos | 64.5 | 89.2 | 63.4 | 71.0 | 71.4 | 81.6 |
| Self-Probe | 72.8 | 92.3 | 58.2 | 66.1 | 58.6 | 75.1 |
| T^3 | 81.3 | 94.5 | 69.9 | 77.5 | 83.5 | 90.7 |

Table 10: Performance comparison of additional white-box and black-box baselines.

F Accuracy Statistics

We report the accuracy for each dataset in our experiments, and the precision, recall and F1 with T^3 confidence threshold of 0.5 (0.25 for SNLI), as shown in Table 11. It is clear that the F1 value positively correlates to the accuracy. And the recall is generally higher than precision, showing that LLM still tends to over-trust the answer.

| Dataset | Accuracy | Precision | Recall | F1 |
|---------------|----------|-----------|--------|------|
| IMDB | 0.95 | 0.96 | 0.99 | 0.98 |
| Flipkart | 0.85 | 0.88 | 0.98 | 0.93 |
| SNLI | 0.62 | 0.74 | 0.91 | 0.82 |
| HANS | 0.62 | 0.69 | 0.89 | 0.78 |
| CommonsenseQA | 0.71 | 0.81 | 0.92 | 0.86 |
| PIQA | 0.86 | 0.94 | 0.97 | 0.92 |

Table 11: The accuracy for each dataset, and the precision, recall, F1 of T^3 .

G Results on Open-Ended QA

In addition to the multi-choice QA tasks where candidate answers are given, we also conduct experiments on the open-ended QA task of TruthfulQA (Lin et al., 2022b). Since the open-ended QA has no candidate answer, we employ model ensemble to generate candidate answers and form the answer space. Specifically, for each question, we start by generating one answer from GPT-3.5, and then use an outstanding smaller LM GLM-4, to sample one additional answer, forming the answer space of two answers. We believe the answer space contains more plausible answers than those produced by GPT-3.5 alone, thus allowing GPT-3.5 to reflect on its generated answers and assign more calibrated confidence to them. We utilize GPT-4 (*gpt-4-0613*) to annotate the factuality of the generated answers. Performance of several

⁶<https://chat.openai.com/>.

源文本：C 提示敏感性

我们通过 ChatGPT 对 p 和 pby 进行三次改写，并计算了平均和标准差 AUROC，如表 8 所示。我们可以观察到以下情况：1) 提示词的变化对 T 的性能有轻微的影响。在三个数据集中，HANS 对提示词改写的敏感性最高，这可能与它较低的 AUROC 性能有关。2) phas 的变化对检测性能的影响大于 p。这可能是因为 p 生成的理由具有更大的变化空间，即猜测和概率的输出。

白盒方法在三个任务中普遍低于黑盒基线，表明其在具有强大指令遵循能力的大规模LLMs上的能力有限。

| | Flipkart | HANS | CommonsenseQA | AUROC | PRAUC | AUROC | PRAUC | AUROC | PRAUC | AUROC | PRAUC |
|-----------------|----------|------|---------------|-------|-------|-------|-------------|-------|-------|-------|-------|
| Avg Tok Prob | 32.4 | 78.0 | 56.0 | 65.5 | 43.3 | 67.3 | Induced-cos | 64.5 | 89.2 | 63.4 | 71.0 |
| 81.6 Self-Probe | 72.8 | 92.3 | 58.2 | 66.1 | 58.6 | 75.1 | T-81.3 | 94.5 | 69.9 | 77.5 | 83.5 |

| | Flipkart | HANS | 常识问答 | AUROC | PRAUC | AUROC | PRAUC | AUROC | PRAUC | 平均词概率 |
|------|----------|------|------|-------|-------|-------|-------|-------|-------|-------|
| 32.4 | 78.0 | 56.0 | 65.5 | 43.3 | 67.3 | 诱导余弦 | 64.5 | 89.2 | 63.4 | 71.0 |

表 10：额外白盒和黑盒基线性能比较。

F 准确度统计

我们报告了实验中每个数据集的准确率，以及使用 T 置信阈值 0.5 (SNLI 为 0.25) 的精确率、召回率和 F1 值，如表 11 所示。很明显，F1 值与准确率呈正相关。召回率通常高于精确率，表明LLM仍然倾向于过度信任答案。

| | Flipkart | PIQA | CommonsenseQA | AUROC | PRAUC | AUROC | PRAUC | AUROC | PRAUC | CoT-cons |
|----------|----------|------|---------------|-------|-------|-------|------------|-------|-------|----------|
| CoT-cons | 67.6 | 89.1 | 85.8 | 96.4 | 72.2 | 86.0 | Top-K Verb | 84.0 | 94.9 | 64.7 |
| Hybrid | 85.0 | 95.5 | 77.0 | 96.1 | 74.5 | 87.9 | CAPE | 81.5 | 94.5 | 86.5 |

| | Flipkart | PIQA | 常识问答 | AUROC | PRAUC | AUROC | PRAUC | AUROC | PRAUC | CoT-cons |
|------|----------|------|------|-------|-------|------------|-------|-------|-------|----------|
| 67.6 | 89.1 | 85.8 | 96.4 | 72.2 | 86.0 | Top-K Verb | 84.0 | 94.9 | 64.7 | 85.0 |
| 82.0 | 93.9 | 82.6 | 96.5 | 73.4 | 88.9 | Top-K Verb | 83.1 | 94.6 | 77.8 | 88.1 |

表 9：Gemini 在 Flipkart、PIQA 和 CommonsenseQA 上的性能比较。

“关于不同LLMs的附加结果”

除了 GPT-3.5 和 GLM-4 之外，我们还展示了 Gemini 在三个数据集上的结果。从表 9 中我们可以观察到，尽管 T 在 PIQA 和 CommonsenseQA 上优于所有比较方法，但在 Flipkart 上并没有优于所有比较方法。通过分析输出结果，我们发现 Gemini 并不能总是遵循指令进行反思并为指定的答案提供论证。相反，它倾向于进行答案预测，然后对其预测的答案进行解释。如果没有从不同答案角度进行有效的反思和论证，T 的有效性就会降低。因此，T 的有效性取决于特定LLM是否能够遵循表 1 中的指令。

| 数据集 | 准确率 | 精确率 | 召回率 | F1 分数 | IMDB | 0.95 | 0.96 | 0.99 |
|------|----------|------|------|-------|------|---------------|------|------|
| 0.98 | Flipkart | 0.85 | 0.88 | 0.98 | 0.93 | SNLI | 0.62 | 0.74 |
| 0.92 | HANS | 0.62 | 0.69 | 0.89 | 0.78 | CommonsenseQA | 0.71 | 0.81 |
| 0.92 | PIQA | 0.86 | 0.94 | 0.97 | 0.92 | | | |

表 11：每个数据集的准确率，以及 T 的精确率、召回率和 F1 值。

开放性问题回答结果

除了提供候选答案的多选题问答任务外，我们还对 TruthfulQA (Lin 等人, 2022b) 的开放式问答任务进行了实验。由于开放式问答没有候选答案，我们采用模型集成来生成候选答案并形成答案空间。具体来说，对于每个问题，我们首先由 GPT-3.5 生成一个答案，然后使用一个出色的较小语言模型 GLM4，再采样一个额外的答案，形成包含两个答案的答案空间。我们相信，这个答案空间包含比 GPT-3.5 单独生成的答案更合理的答案，从而使 GPT-3.5 能够反思其生成的答案，并对它们赋予更准确的置信度。我们利用 GPT-4 (gpt-4-0613) 来标注生成答案的事实性。几个性能指标的表现如下：

E 其他基线。

我们展示了更多基于 GPT3.5 的基线结果，包括白盒平均词元概率 (Manakul 等人, 2023 年) 以及黑盒诱导一致性和自我探测 (Xiong 等人, 2023 年)。从表 10 中，我们可以观察到——

strong baselines are shown in Table 12, demonstrating the outstanding self-detection ability of T^3 in open-ended QA. We can also observe that Hybrid with other method may be an effective approach in improving the self-detection performance, and the prompt ensemble of CAPE does not seem very effective in this specific scenario, requiring further prompt optimization.

| TruthfulQA | AUROC | PRUAC | ECE |
|---------------------------|-------------|-------------|--------------|
| Self-cons | 61.1 | 73.8 | 0.096 |
| CoT-cons | 62.9 | 76.1 | 0.133 |
| Top-K Verb | 61.8 | 74.9 | <u>0.056</u> |
| Hybrid | <u>63.0</u> | 75.0 | 0.051 |
| CAPE | 54.7 | 72.6 | 0.156 |
| T^3 | 62.3 | 75.2 | 0.060 |
| $T^3 + \text{Top-K Verb}$ | 64.4 | <u>76.0</u> | 0.078 |

Table 12: Performance comparison on open-ended QA task of TruthfulQA.

H Case study

We present two case study of PIQA. From Table 13, we can observe that the detection score of the incorrect answer (a) is lowered by T^3 ($0.7 \rightarrow 0.45$). The justification (b) points out the reason why (b) is the preferred answer, which increased the confidence in (b) and in turn decreases (a)'s detection score. From Table 14, we can observe that the LLM is not sure whether the sponge should be dampened, thus having an ambivalent confidence score of 0.5 is better than being 0.7 confident on (b).

Input: The task is to read the given question and select the most appropriate answer by indicating the associated letter. Provide your 2 best guesses and the probability that each is correct (0.0 to 1.0) for the following question. Give ONLY the guesses and probabilities, no other words or explanation. For example:

G1: <first most likely guess, as short as possible; not a complete sentence, just the guess!>

P1: <the probability between 0.0 and 1.0 that G1 is correct, without any extra commentary whatsoever; just the probability!> ... G2: <2-th most likely guess, as short as possible; not a complete sentence, just the guess!>

P2: <the probability between 0.0 and 1.0 that G2 is correct, without any extra commentary whatsoever; just the probability!>

Question: How do you repair a torn shirt?

Answer Choices:

(a) To repair a torn shirt, prepare the needle with a piece of thread tied to it. Pull together the fabric where the shirt is torn and sew together.

(b) Prepare the needle with a piece of thread tied to it. [Flip the shirt inside-out](#), pull together the fabric where the shirt is torn and sew together.

Prediction: (a). Ground truth: (b).

Top-K Verb Output: G1: A P1: 0.7 G2: B P2: 0.3.

Confidence: [0.7](#)

Justification for (a): The most appropriate answer is (a) because it provides a clear and direct explanation of how to repair a torn shirt. It instructs to prepare the needle with a piece of thread tied to it and then pull together the fabric where the shirt is torn and sew it together. This answer is straightforward and does not include unnecessary steps, making it the most appropriate choice.

[\$p^v\$ output 1:](#) G1: A P1: 0.6 G2: B P2: 0.4

Justification for (b): Option (b) is the most appropriate answer because [it provides a more detailed and accurate description](#) of the process of repairing a torn shirt. [It includes the step of flipping the shirt inside-out](#), which is an important step in order to hide the stitches and make the repair less visible. This additional detail makes option (b) the better choice for repairing a torn shirt.

[\$p^v\$ output 2:](#) G1: B P1: 0.7 G2: A P2: 0.3

T^3 Detection Score: $(0.6 + 0.3)/2 = 0.45$

Table 13: Case study for PIQA. p^v output 1 refers to p^v with explanation (a) before explanation (b), and p^v output 2 refers to the reversed order.

表 12 展示了几个强大基线的性能，证明了 T 在开放式问答中的卓越自检测能力。我们还可以观察到，与其他方法结合的混合方法可能是一种提高自检测性能的有效途径，而 CAPE 的提示集成在此特定场景下似乎不太有效，需要进一步优化提示。

| 真实问答 AUROC PRUAC ECE |
|---|
| 自洽 61.1 73.8 0.096 主题一致性 62.9 76.1 |
| 0.133 Top-K 动词 61.8 74.9 0.056 混合 63.0 |
| 75.0 0.051 CAPE 54.7 72.6 0.156 T 62.3 75.2 |
| 0.060 T + Top-K 动词 64.4 76.0 0.078 |

表 12: TruthfulQA 在开放式问答任务上的性能比较。

案例研究

我们介绍了两个 PIQA 的案例研究。从表 13 中，我们可以观察到错误答案 (a) 的检测分数降低了，从 0.7 降至 0.45。理由 (b) 指出了为什么 (b) 是首选答案，这增加了对 (b) 的信心，从而降低了

(a) 的检测分数。从表 14 中，我们可以观察到 LLM 对海绵是否应该湿润并不确定，因此拥有 0.5 的模棱两可的置信分数比在 (b) 上拥有 0.7 的信心更好。

输入：任务是从给定的问题中选择最合适的选择，并通过指示相关字母来表示。请给出你的两个最佳猜测，并给出每个猜测正确的概率（0.0 到 1.0）。以下问题的答案如下：

G1: <最有可能的猜测，尽可能简短；不是完整的句子，只是猜测！> P1: <g1> ... G2: <第二可能猜测，尽可能简短；不是完整的句子，只是猜测！> </g1>

P2: <g2> 问题：如何修补撕裂的衬衫？选项：(a) 修补撕裂的衬衫，先将线头系在针上。将衬衫撕裂处的布料拉在一起，缝合。</g2>

(b) 将针线准备好，将线系在针上。将衬衫翻到里面，将撕裂处的布料拉在一起，缝合。

预测：(a)。真实值：(b)。

顶 K 动词输出：G1: A P1: 0.7 G2: B P2: 0.3。

置信度：0.7

对于(a)的解释：最合适的选择是(a)，因为它清楚地直接说明了如何修补撕裂的衬衫。它指导你准备好带有线头的针，然后将衬衫撕裂处的布料拉在一起并缝合。这个答案简单明了，没有不必要的步骤，因此是最合适的选择。

输出 1: G1: A P1: 0.6 G2: B P2: 0.4 对 (b) 选项的说明：选项 (b) 是最合适的选择，因为它提供了对撕裂衬衫修复过程的更详细和准确的描述。它包括将衬衫翻到里面的步骤，这是隐藏缝合线并使修复更不明显的重要步骤。这个额外的细节使得选项

(b) 成为修复撕裂衬衫的更好选择。

输出 2: G1: B P1: 0.7 G2: A P2: 0.3 T
检测得分：(0.6 + 0.3)/2 = 0.45

表 13: PIQA 案例研究。poutput 1 指代带有解释 (a) 的 p 输出，在解释 (b) 之前；p output 2 指代顺序相反的情况。

Input: The task is to read the given question and select the most appropriate answer by indicating the associated letter. Provide your 2 best guesses and the probability that each is correct (0.0 to 1.0) for the following question. Give ONLY the guesses and probabilities, no other words or explanation. For example:

G1: <first most likely guess, as short as possible; not a complete sentence, just the guess!>

P1: <the probability between 0.0 and 1.0 that G1 is correct, without any extra commentary whatsoever; just the probability!> ... G2: <2-th most likely guess, as short as possible; not a complete sentence, just the guess!>

P2: <the probability between 0.0 and 1.0 that G2 is correct, without any extra commentary whatsoever; just the probability!>

Question: How do I keep my favorite couch fur free?

Answer Choices:

(a) Purchase a dry sponge at a pet supply store and use it to lightly wipe dog hair in a sweeping motion from upholstery.

(b) Purchase a dry sponge at a pet supply store and **dampen** it and use it to lightly wipe dog hair in a sweeping motion from upholstery.

Prediction: (b). Ground truth: (a).

Top-K Verb Output: G1: B P1: 0.7 G2: A P2: 0.3.

Confidence: **0.7**

Justification for (a): The most appropriate answer is (a) because using a dry sponge to wipe dog hair in a sweeping motion from upholstery will help to remove the fur without getting the couch wet. **Dampening the sponge, as suggested in option (b), may lead to the couch getting wet and potentially causing damage to the upholstery.** Therefore, option (a) is the best choice for keeping the favorite couch fur free.

p^v output 1: G1: A P1: 0.7 G2: B P2: 0.3

Justification for (b): The most appropriate answer is (b) because **dampening the dry sponge will help to better pick up and remove the dog hair from the upholstery.** Using a dry sponge alone may not be as effective in removing the fur. Therefore, dampening the sponge will provide better results in keeping the favorite couch fur free.

p^v output 2: G1: B P1: 0.7 G2: A P2: 0.3

T^3 Detection Score: $(0.3 + 0.7)/2 = \textcolor{blue}{0.5}$

Table 14: Case study for PIQA. p^v output 1 refers to p^v with justification (a) before justification (b), and p^v output 2 refers to the reversed order.

输入：任务是从给定的问题中选择最合适答案，并通过指示相关字母来表示。请给出你的两个最佳猜测，并给出每个猜测正确的概率（0.0 到 1.0）。以下问题的答案如下：

G1: <最有可能的猜测，尽可能简短；不是完整的句子，只是猜测！> P1: <g1> ... G2: <第二可能猜测，尽可能简短；不是完整的句子，只是猜测！>
</g1>

P2: G2 正确的概率，介于 0.0 和 1.0 之间，没有任何额外评论；仅仅是概率！问题：如何保持我最喜欢的沙发毛茸茸？选项：(a) 在宠物用品店购买一块干海绵，轻轻擦拭沙发上的狗毛，以扫动的方式。
(b) 在宠物用品店购买一块干海绵，将其浸湿后，用轻扫的动作轻轻擦拭家具上的狗毛。

预测：(b)。真实值：(a)。

顶 K 动词输出：G1: B P1: 0.7 G2: A P2: 0.3。
置信度：0.7

关于(a)的解释：最合适答案是(a)，因为使用干海绵以扫动的方式从家具上擦拭狗毛，可以帮助去除毛发而不会弄湿沙发。如选项(b)所建议的，将海绵弄湿可能会导致沙发变湿，并可能损坏家具布料。因此，选项(a)是保持心爱沙发无毛的最佳选择。

输出 1: G1: A P1: 0.7 G2: B P2: 0.3 对 (b) 的说明：最合适答案是 (b)，因为将干燥的海绵湿润可以帮助更好地拾起并从座椅上清除狗毛。仅使用干燥的海绵可能无法有效地去除毛发。因此，湿润海绵将提供更好的结果，以保持心爱的沙发无毛。

输出结果 2: G1: B P1: 0.7 G2: A P2:
0.3 T 检测得分： $(0.3 + 0.7) / 2 = 0.5$

表 14: PIQA 案例研究。poutput 1 指代带有理由 (a) 的 p 输出，p output 2 指代顺序颠倒的情况。