Quantifying Genuine Awareness in Hallucination Prediction Beyond Question-Side Shortcuts

Anonymous ACL submission

Abstract

Many works have proposed methodologies for language model (LM) hallucination detection and reported seemingly strong performance. However, we argue that the reported performance to date reflects not only a model's genuine awareness of its internal information, but also awareness derived purely from question-side information (e.g., benchmark hacking). While benchmark hacking can be effective for boosting hallucination detection score on existing benchmarks, it does not generalize to out-of-domain settings and practical usage. Nevertheless, disentangling how much of a model's hallucination detection performance arises from question-side awareness is non-trivial. To address this, we propose a methodology for measuring this effect without requiring human labor, Approximate Question-side Effect (AQE). Our analysis using AQE reveals that existing hallucination detection methods rely heavily on benchmark hacking. The code is available online (https://anonymous.4open.science/r/AQE-0717).

1 Introduction

The defining and quantifying of human-like mental attributes in large language models (LLMs) lie at the heart of a long-standing question: whether artificial systems can possess minds akin to our own. While recent advances show that LLMs can rival or even surpass humans in rational reasoning tasks (Brown et al., 2020; Grattafiori et al., 2024; OpenAI, 2024), higher-order traits such as self-awareness and emotion remain poorly understood, partly due to ambiguities in their definition and measurement (Li et al., 2024b; Yin et al., 2023).

Among these traits, self-awareness of knowledge is particularly important because of its close connection to hallucination detection, which is critical for the reliability of LLMs. Although hallucination can arise from various sources, a major cause is answering questions beyond the model's pre-trained knowledge (Tonmoy et al., 2024). Humans can recognize when they lack relevant knowledge and refrain from answering (Irak et al., 2019; Koriat, 1993), whereas LLMs lack such awareness and tend to generate plausible outputs regardless, leading to hallucination.

Then, how can we define and measure self-awareness of LLMs? Prior work has often equated self-awareness with hallucination detection itself, motivated by its practical importance. Indeed, recent studies report high hallucination detection performance (Snyder et al., 2024; Zhang et al., 2024; Manakul et al., 2023; Azaria and Mitchell, 2023).

However, we argue that hallucination prediction does not directly measure self-awareness, because two distinct sources of information are typically involved in the prediction process: (1) information about the model itself and (2) information about the question. As such, hallucination prediction reflects a mixture of **model-awareness (self-awareness)** and **question-awareness**. To isolate self-awareness, we disentangle these two components and introduce a Shapley-based metric, the Approximate Question-side Effect (AQE), to quantify question-awareness. The contribution of self-awareness is then estimated by subtracting AQE from hallucination detection precision.

Quantifying self-awareness has important practical implications. As shown in §4.2, hallucination predictors that rely heavily on question-awareness often exploit dataset-specific shortcuts and fail to generalize under distribution shifts. In contrast, approaches grounded in model-side information yield more robust performance. We empirically support this claim through dataset analyses and experiments in §B, §4.3, and §5.

Lastly, we also propose a method to enhance the use of model-side information, by leveraging the confidence scores of LLMs more effectively. The proposed method is **S**emantic **C**ompression by **A**nswering in **O**ne word (SCAO). We demonstrate

that SCAO performs particularly well in low AQE settings. Though this method shows limitations in certain settings, such as long-form question answering, it provides clues to overcoming the limitations of previous probing-based methods.

Our contributions are summarized as follows:

- Conceptual: We disentangle hallucination detection into self-awareness and question-awareness and provide a measurable definition of self-awareness in LLMs.
- Methodological: We introduce AQE, a Shapley-based metric to quantify question-side effects.
- Empirical: We show that shortcut-driven, question-aware methods fail to generalize, while model-side approaches are more robust.

2 Related works

As human self-awareness has been extensively studied in cognitive psychology and neuroscience, its core mechanisms can be leveraged to structure and categorize approaches for evaluating internal confidence and hallucination in LLMs.

Self-awareness in humans: Insights from cognitive neuropsychology Extensive research in cognitive psychology and neuroscience has shown that human self-awareness—particularly in the context of knowing whether one knows something—relies on layered cognitive processes. According to studies such as (Koriat, 1993; Irak et al., 2019; Brown et al., 2017), two key mechanisms underpin this self-assessment.

1) Unconscious level: When a query is received, the brain initiates implicit memory retrieval, evaluating whether candidate memories fit the contextual cues. This process activates regions such as the orbitofrontal and prefrontal cortices within 300–500 milliseconds (Schnider, 2001; Irak et al., 2019), distinct from areas responsible for linguistic output, such as the posterior temporal lobe (i.e., Wernicke's area) (Binder, 2015). Several factors—such as the amount, accessibility, and vividness of retrieved information—act as internal cues signaling potential knowledge (Koriat, 1993).

2) Conscious level: The results of these unconscious processes are then consciously evaluated through metacognitive strategies. These include checking for logical and temporal consistency or aligning retrieved information with known frameworks. This layered evaluation reflects human self-awareness—our capacity to introspect on our own knowledge states and confidence levels.

The dual-process theory (Kahneman, 2011) offers a broader framing: rapid, intuitive processes dominate simple recall tasks, while complex, deliberative processes support tasks such as reasoning or problem-solving.

Self-awareness in LLMs: A perspective on hallucination detection In large language models (LLMs), the concept of self-awareness can be reinterpreted as the model's ability to internally assess whether it possesses sufficient knowledge to answer a question accurately. This introspective capacity—whether performed before or after answer generation—is directly related to hallucination detection mechanisms. Our analysis proposes aligning hallucination mitigation strategies with the structure of human self-awareness processes.

1) Before-generation: Approaches that attempt to detect potential hallucinations *before* the model generates an answer mirror the unconscious processes of humans. These methods, including our own, rely on internal indicators such as activation patterns or uncertainty proxies to determine knowledge sufficiency prior to verbalization (Mallen et al., 2022). Benchmarks like Mintaka (Sen et al., 2022) and ParaRel (Elazar et al., 2021) primarily test immediate factual recall, aligning well with this layer of self-assessment.

2) After-generation: Other approaches evaluate the model's response **after** generation, resembling conscious-level reasoning in humans. These include multi-pass generation, self-consistency checks, and the integration of external tools such as retrievers (Béchard and Ayala, 2024; Manakul et al., 2023; Chen et al., 2024). Benchmarks requiring structured reasoning, such as MMLU (Hendrycks et al., 2021), TruthfulQA (Lin et al., 2022), and ELI5 (Fan et al., 2019), are more aligned with this stage, as they demand deliberative thought processes rather than pure retrieval.

Importantly, not all hallucinations can be attributed to failures in self-awareness. For instance, hallucinations in open-book QA tasks may arise from comprehension or inference errors rather than epistemic uncertainty. Benchmarks like SQuAD (Rajpurkar et al., 2016) and FEVER (Thorne et al., 2018) exemplify this issue, being more about information grounding than self-assessment.

Overall, hallucination in LLMs encapsulates a variety of cognitive failures. Addressing them requires different forms of internal awareness—ranging from assessing memory sufficiency to verifying logical coherence. Ultimately, a robust system may need to combine multiple self-assessment mechanisms. In this work, we focus on introspective strategies relevant to knowledge recall before answer generation, drawing inspiration from unconscious-level self-awareness in humans.

3 Definition

In this section, we first examine the definition of self-awareness of human. Next, we define the task formulation of hallucination prediction, to establish a definition of self-awareness in LLMs. And we review the definitions from previous works. More detailed review on the related works of neuropsychology and LLMs is in §2.

Self-awareness of human In psychology, human self-awareness is defined as the capability of perception of one's own mental processes or states, which includes thoughts, feelings, emotions, and knowing (Morin, 2011). In this work, we focus on the self-awareness of knowing certain knowledge, which is also referred to as self-knowledge (Yin et al., 2023).

Some studies on LLMs also borrow the term "meta-cognition" from psychology to refer to self-knowledge (Li, 2023), which may be inaccurate. It is because meta-cognition focuses on the conscious level (Koriat, 1993), while recent research suggests that the human brain conducts judgment of knowing even at the unconscious level (Irak et al., 2019), which is even before consciously recognizing the meaning of the question.

Hallucination prediction The term "hallucination" has been broadly used to refer to the phenomenon where a model provides an incorrect answer to a given question (Li et al., 2023; Manakul et al., 2023; Béchard and Ayala, 2024). Thus, "hallucination detection" refers to the task of predicting whether a response is incorrect (Li et al., 2023; Chen et al., 2024).

In this work, we focus specifically on (1) hallucination from **factoid questions** that examine whether the model possesses certain knowledge, as this has been widely used as a clear and straightforward scenario for exploring hallucination detection (Snyder et al., 2024; Zhang et al., 2024). (2) And we focus on an early detection (i.e., **prediction**) scenario (Snyder et al., 2024; Chen et al., 2024; Azaria and Mitchell, 2023) where k is predicted before answer generation, as this setting is more appropriate for examine self-awareness (we describe this in the next section).

To formulate the common problem setting of hallucination prediction, let θ represent the model, x the query, and y the answer label. The model θ infers $\hat{y}$ from the input x. The correctness of $\hat{y}$ can then be measured by evaluating its similarity to y, denoted as k, representing a binary value (True/False). Common evaluation methods include string matching (Zhang et al., 2024), GLUE, and G-eval (Liu et al., 2023). During this process, a datapoint $s_{\theta,x} = \theta(x)$ is extracted from θ, which contains information about how θ perceived x. We denote this as s for simplicity. Through a series of question-answering and evaluation processes, we obtain a dataset $\mathcal{D} = \{(s_i, k_i)\}_{i=1}^{N}$, where N is the dataset size. From this, hallucination prediction is defined as a binary classification task where a learnable module ϕ learn to take input s_i to predict k_i, which we note $\hat{k_i} = \phi(s_i)$.

Assuming θ as a transformer model (Vaswani et al., 2023), s can be mainly two forms (1) Hidden state vectors: Transformer models are composed of multiple attention layers, where each layer passes a fixed-size vector (i.e., hidden states) to the next. These vectors encode the semantic and contextual information of the input x (Reimers and Gurevych, 2019), and are also known to contain information about the model's response that will be generated (Li et al., 2024a), providing a cue for hallucination prediction. (2) Confidence score: This refers to the softmax probability that the causal LLMs predict for the next token generation. While the hidden state is a high-dimensional representation (4096 dimensions in `LLaMA-3-8B`), the confidence score is a scalar value. As the hidden state contains richer information, it has achieved the best performance as a source for hallucination prediction and has been regarded as the main source (Snyder et al., 2024; Chen et al., 2024).

3.1 Formulating the self-awareness of LLM

We defined hallucination prediction as the process where ϕ perceives s to infer $\hat{k}$ to predict k. As s represents the state of the model after it has seen a question, it inherently contains information of two distinct objects, the question-side and the model-side, as illustrated in Figure 1. The

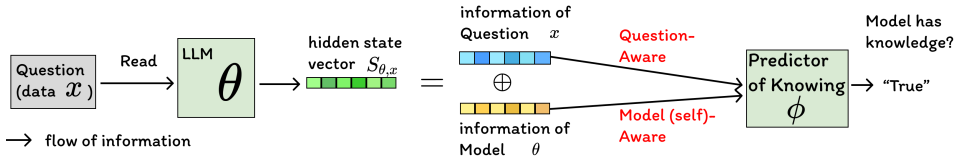

Figure 1: Pipeline for the prediction of knowing (prediction of hallucination).

question-side refers to the objective information that can be shared across different models, such as the domain of question (e.g., science, math, society) and the type of question (e.g., multiple choice, open-ended). For human, this type of information is derived from rational abilities (e.g., classification and reading comprehension), rather than from higher-order mental capacity such as self-awareness (Morin, 2011). In contrast, the model-side information refers to model-specific attributes, such as the possession of the knowledge that is needed for response, or the degree of confidence for answering. In humans, this corresponds to the domain of self-awareness.

Let us denote the representation of the question-side information in s as s_Q, and the representation of the model-side information as s_M. We rewrite previously defined hallucination prediction $\hat{k} = \phi(s)$ as $\hat{k} = \phi(s_Q, s_M)$. When we denote the information contained in s as s itself, we can also note $s = s_Q \cup s_M$. This decomposition forms the basis for applying the Shapley value formulation.

Prior studies have also empirically shown that the hidden states of transformer models encode multiple properties in a linearly separable manner. For instance, Li et al. (2024a) demonstrates that the hidden state contains a direction associated with the attribute of "truthfulness", therefore linearly adding this to the hidden states results in more truthful responses.

When ϕ learns to predict k using two sources of information (s_Q, s_M), utilizing question-side information can be regarded as question-awareness, while utilizing model-side information can be regarded as model-awareness, which is "self-awareness" in the perspective of model. Thus, self-awareness can be formally expressed as:

$$\hat{k} = \phi(s_M). \quad (1)$$

Why hallucination prediction, not detection? We argue that hallucination prediction is a more suitable setting than hallucination detection for examining self-awareness. In hallucination detection, ϕ perceives model-generated answers when predicting $\hat{k}$, which can be formulated as $\hat{k} = \phi(s_M, s_Q, x, \hat{y})$, where x is the question and $\hat{y}$ is the generated answer. As self-awareness is defined as $\hat{k} = \phi(s_M)$, additional inputs x and y serve as distracting factors, making it difficult to isolate the effect of s_M. Intuitively, this detection scenario may become more of a reading comprehension task over x and y, rather than assessing the model's internal states. Therefore, we choose hallucination prediction scenario to more clearly examine the effect of self-awareness.

3.2 Definition from previous works

Utilizing s as self-awareness Previous works implicitly regard self-awareness as the hallucination detection itself. In other words, the focus has been on the act of predicting k from s, with no consideration given to the decomposition of s into s_Q and s_M. As a result, some of the hallucination detection performance reported in those works is partially overestimated by the effect of question-side shortcuts (Zhang et al., 2024; Azaria and Mitchell, 2023). We analyze such cases in §4.1.

Utilizing s_Q as self-awareness In another case, some works define utilization of s_Q as self-awareness, which is the opposite concept from our view. (Yin et al., 2023) defines the term "self-knowledge" as a self-awareness on possession of certain knowledge. And from this definition, they construct the dataset SelfAware to measure the self-awareness capacity. While the definition in the paper is consistent with ours, the construction of the dataset stands on the opposite definition.

The dataset SelfAware consists of "answerable" and "unanswerable" questions, and the capacity of self-awareness is defined as the classification between the two. An unanswerable question refers to one that is philosophical (e.g., "What is a happy life?") or subjective ("Do you like to go to the mountains?"), where no definitive answer can be given, thus inevitably leading to hallucination. This setting is contradicting the term "self-awareness" in three points: (1) the unanswerability defined in the SelfAware involves no model-side information. For example, the question "Do you like to go to the mountains?" is always classified as "unanswerable"

in SelfAware, regardless of how much knowledge about mountains is stored in the answerer (e.g., 1B LM, 70B LM, or a human).

(2) The dataset includes fixed labels indicating the unanswerability of each question, therefore unanswerability is entirely independent from the model's state. As unanswerability represents whether the model can answer a given question, it aligns with k in our problem setting (Equation 1). However, as k in SelfAware is determined entirely by properties of the question and not the model's state, the task proposed in this dataset becomes $k = \phi(s_Q)$, excluding s_M. This problem setting is a question-awareness, not a self-awareness, from our definition. (3) For humans, the ability needed for this classification task is reading comprehension, which is not self-awareness.

4 AQE: assessing question-side effects of hallucination prediction datasets

As question-awareness is using and relying on the question-side information, in this section, we first identify the data-specific shortcuts that cause dependency on the question-side information, through a case study in existing datasets for hallucination prediction. Next, we introduce our novel metric AQE, a method for quantifying the effect of question-side shortcuts in that dataset.

4.1 Case study on question-side shortcuts

We investigate sources of question-side shortcuts in datasets used in hallucination prediction studies. We focus on short-form closed-book factoid question answering scenarios, which include ParaRel (Elazar et al., 2021), Mintaka (Sen et al., 2022), HaluEval (Li et al., 2023), HotpotQA (Yang et al., 2018), and SimpleQuestion (Yin et al., 2016). We describe three main sources of question-side shortcuts, and further in §B.

(1) Broken Question Problem Many datasets have incomplete annotations for one-to-many question–answer pairs (e.g., Daniel Bernoulli → only "physics" labeled). This causes correct answers outside the label set to be marked as hallucinations, making the predictor ϕ learn domain-based shortcuts rather than true self-awareness. Datasets like Mintaka fix this by adding constraints for one-to-one mappings.

(2) Domain Shortcut The likelihood of hallucination (k=True) often varies by domain. For instance, if a model is strong in science but weak in history, history questions will naturally show higher hallucination rates. Thus, ϕ may simply learn domain-based correlations — predicting (k) based on what the question is about rather than what the model actually knows. This makes the task "question-aware" instead of "model-aware," undermining the goal of self-awareness. A truly self-aware model should, like a human, recognize when it specifically knows or doesn't know something, even within an unfamiliar domain.

(3) Question Type Shortcut The structure of the question itself (e.g., binary-choice, multiple-choice, open-ended) also influences (k). Binary-choice questions have a higher baseline chance of being correct $p(k = True) \geq 0.5$ even with random guessing, unlike open-ended ones $p(k = True) = 0$. Consequently, ϕ might exploit this and always predict "True" for binary-choice questions, forming another non-self-aware shortcut. Datasets such as HotpotQA, HaluEval, and Mintaka contain such biases.

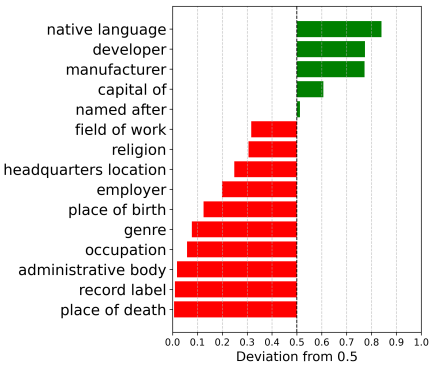

Figure 2: The portion of $k = True$ for each domain, by `LLaMA-3-8B` model on the ParaRel train set. The rate is skewed toward 0 or 1 by domain, rather than being centered around 0.5.

There are various other question-side shortcuts, which are described in the §B. These shortcuts can be identified by considering various scenarios in which they may act as shortcuts, and it is likely that some shortcuts remain undiscovered, as it is very subtle to determine. Therefore, manually identifying and removing them from datasets is nontrivial. That is why we introduce AQE in the next section, a method for approximately assessing the total effect of question-side shortcuts without human investigation.

4.2 Approximate question-side effect

In this section, we describe the concept of AQE. Repeating §3, the model θ is given a question x

to generate answer, where the correctness is denoted as a binary representation k. A hidden state s is extracted at the first position of the answer. And s is consist of model-side s_M and question-side information s_Q. Learnable module ϕ can perceive s to infer $\hat{k}$ to predict k, which is hallucination prediction. Self-awareness is defined as utilizing s_M to predict k. Here, what we ultimately aim to measure is the effect of utilizing s_M in predicting k, denoted as $\mathcal{A}(\phi(s_M))$. $\mathcal{A}(\cdot)$ is a metric that measures the correctness of the predicted $\hat{k}$. As all we can measure is $\mathcal{A}(\phi(s_Q, s_M))$, we decompose this as follows: $\mathcal{A}(\phi(s_Q, s_M)) \approx \mathcal{A}(\phi(s_Q)) + \mathcal{A}(\phi(s_M))$ This allows us to measure the effect of self-awareness as follows:

$$\mathcal{A}(\phi(s_M)) \approx \mathcal{A}(\phi(s_Q, s_M)) - \mathcal{A}(\phi(s_Q)) \quad (2)$$

AQE as a Shapley analysis We formalize AQE as a special case of Shapley analysis (Fryer et al., 2021), which evaluates the impact of individual factors on the outcome. Specifically, AQE corresponds to the concept of **marginal contribution**—a metric that quantifies the separate contribution of a single factor.

$$\Gamma_\beta(\alpha) = \Gamma(\alpha \cup \beta) - \Gamma(\beta) \quad (3)$$

Equation 3 is the general formulation of marginal contribution. Here, α and β represent individual components of a system. $\alpha \cup \beta$ represents a state in which these components are mixed together. $\Gamma(\cdot)$ represents the baseline performance metric (e.g., AUROC). $\Gamma_\beta(\alpha)$ quantifies the impact of removing β, thus isolating the contribution of α. In our setting, α and β correspond to the model-side and question-side information, respectively. $\alpha \cup \beta$ can be interpreted as the information contained in the hidden state of the model, which integrates both the model and question side information. $\Gamma(\alpha \cup \beta)$, $\Gamma(\beta)$, and $\Gamma_\beta(\alpha)$ correspond to $A(\phi(s))$, $A(\phi(s_Q))$, and $A(\phi(s_M))$, respectively.

Computing of self-awareness Computing $A(\phi(s_Q))$ is achieved by introducing a distinct model θ' (where $\theta' \neq \theta$) which is optimized to only embed basic properties of the input question (e.g., domain or question type), $s'_Q \approx s' = \theta'(x)$. A representative example of θ' is sBERT (Reimers and Gurevych, 2019). sBERT is a very small model with only 22.7M parameters, but it is optimized to generate an embedding vector s' from input text x (e.g., question). sBERT is known to capture high-level information from text as effectively as θ with a larger architecture (e.g., `LLaMA-3-8B`), as long as the target information is simple enough (e.g., domain classification). Therefore, while s'_Q and s_Q reside in different representational spaces of two distinct models, they are assumed to capture similar high-level information ($s'_Q \sim s_Q$). Conversely, due to its small size, we can assume θ' holds a very small amount of knowledge, which makes the knowledge distribution of θ' and θ independent (s'_M and s_M are independent). This makes s'_M ignored when ϕ learning to predict k (correctness of θ) from s' (hidden state from θ').

Table 1: AQE assessment on datasets. Prediction of k from `LLaMA-3-8B`, with s' from sBERT.

	short-form					long-form
	ParaRel	Mintaka	HaluEval	HotpotQA	Simple Question	Explain
$p(k = True)$	54.31	55.01	37.51	32.71	19.08	31.63
$p(k = False)$	45.68	44.98	62.48	67.28	80.19	68.36
AQE_{acc}	73.26	63.50	68.55	72.50	82.36	65.65
AQE_{auc}	82.61	66.67	68.37	70.14	68.13	69.40

This results in Equation 4, where ϕ and ϕ' share architecture but are trained independently, as s_Q and s'_Q lie in different representational spaces.

$$\mathcal{A}(\phi(s_Q)) \approx \mathcal{A}(\phi'(s')) \quad (4)$$

The resulting $\mathcal{A}(\phi(s_Q))$ of Equation 4 is defined as AQE. To summarize, θ' predicts k of θ without using information of θ. Intuitively, a distinct model θ' predicts whether θ will succeed on a given question, using only the information from the question. As no information about θ is involved, this setup excludes self-awareness and reflects only question-awareness. Together with Equation 2, we can derive $\mathcal{A}(\phi(s_M))$, the component of self-awareness in the measured hallucination prediction performance: $\mathcal{A}(\phi(s_M)) \approx \mathcal{A}(\phi(s)) - \mathcal{A}(\phi'(s'))$

However, this formulation of AQE holds only under the assumption that s is the hidden state format. When s is a confidence score, AQE cannot be directly applied because confidence score is a one-dimensional value, which is too saturated to embed high-level information of the question.

4.3 Measuring AQE across datasets

In this section, we measure AQE across hallucination prediction datasets. The model θ is `LLaMA-3-8B-Instruct`[1], and we evaluate it mainly on short-form factoid datasets (e.g., ParaRel, Mintaka, Halueval, HotpotQA), and additionally on long-form factoid datasets (e.g., Ex-

[1]https://huggingface.co/meta-llama/Meta-Llama-3-8B

Table 2: AQE score of dataset and `LLaMA-3-8B` model. The version (original, type, domain) with the lowest AQE within each dataset is highlighted in **bold**.

	Mintaka			HotpotQA		ParaRel		Explain	
	original	+type	type + domain	original	+type	original	+domain	original	+domain
$p(k = True)$	55.01	49.71	53.07	42.33	29.12	54.31	60.45	31.63	39.83
$p(k = False)$	44.98	50.28	46.92	62.48	70.87	45.68	39.54	57.66	60.16
AQE_{acc}	63.50	59.81	**59.04**	68.55	**76.03**	73.26	**55.09**	65.65	**61.21**
AQE_{auc}	66.67	64.06	**61.62**	68.37	**55.37**	82.61	**57.55**	69.40	**61.89**

plain). The details of each dataset are provided in the §D. We use two metrics: AUROC and accuracy. AQE for each metric is denoted as AQE_{auc} and AQE_{acc}, respectively. We also report the $p(k = True)$ and $p(k = False)$, the bias of binary label k. We show that a larger (70B) model shows similar trends (§E).

As shown in Table 1, the AQE_{auc} for most datasets is close to or exceeds 0.70, rather than centering around 0.5. This indicates that many datasets exhibit a strong effect of question-side shortcuts, and a model can easily achieve AUROC over 0.70 without any self-awareness (i.e., perception of model-side information), relying solely on question-aware skills such as domain classification. Though previous works (Snyder et al., 2024; Zhang et al., 2024) have reported to achieve AUROC over 0.80 in hallucination prediction using these datasets, it could be a statistical artifact that is hard to generalize. In all datasets, AQE_{acc} is consistently higher than the $p(k = True)$, indicating that AQE captures question-side effect beyond just the bias of the binary label.

4.4 AQE in refined dataset

In §4.1, we analyze that the type and the domain of a question can act as question-side shortcuts in predicting k. Fortunately, some datasets (HotpotQA, Mintaka, and ParaRel) include tags for this information, which allows us to control for it. We analyze AQE for each dataset before and after this control. In Table 10, the "+type" column refers to the dataset after excluding binary-type questions. "+domain" indicates a regrouping of the train and test data according to their categories, such that the domains do not overlap (i.e., out-of-domain setting). The refinement process is detailed in §D.

The experimental results can be summarized in two main points. (1) Applying refinement leads to a significant reduction in AQE. This demonstrates that through post-processing, we can get a dataset with lower dependency on question-side information, which is more suitable for evaluating self-awareness. (2) AQE still remains even after refinement. This suggests the presence of a question-side effect that we have not yet identified or controlled for.

4.5 SCAO

We propose a method called Semantic Compression by Answering in One word (SCAO), which leverages model-side information more effectively by instructing the model to "answer in one word," thereby improving the alignment of confidence values. We provide further explanation in §C.

5 Experiment on hallucination prediction approaches

In this section, we evaluate hallucination prediction approaches across multiple datasets and their refined versions.

Approaches Previous approaches for hallucination prediction can be broadly categorized into three. (1) confidence-based: this utilizes the softmax probability of the answer token (Fadeeva et al., 2024). It is utilized in other forms, such as perplexity (Ren et al., 2023) or energy (Liu et al., 2021). We adopt a simplified method that takes the mean of top-n softmax probabilities of the first answer token and applying a threshold. n and the threshold t are learnable ϕ. (2) hidden-state-based: This approach feeds the hidden-state vectors of a model θ into learnable ϕ. ϕ can be a linear layer (Li et al., 2024a) or more complex architecture (Azaria and Mitchell, 2023; Chen et al., 2024). We adopt a three-layer deep neural network. (3) aggregation: This approach concatenates the confidence scores and hidden state into a single vector, which is then passed to a learnable ϕ (Snyder et al., 2024). In Table 11, *Conf*, *Probe*, and *Conf + Probe* represent the three evaluated approaches, respectively. Detailed explanations are provided in §E.1. We conduct experiments using instruction-tuned LLaMA models of two different sizes (8B, 70B).

Dataset and metric For the evaluation of hallucination prediction, we narrow down our focus

Table 3: Hallucination prediction performance (AUROC) of instruction-tuned 8B and 70B LLaMA models across multiple datasets.

(a) Mintaka

	8B						70B					
	original		+ type		+ type + domain		original		+ type		+ type + domain	
	auroc	$\mathcal{A}(\phi(s_M))$	auroc	$\mathcal{A}(\phi(s_M))$	auroc	$\mathcal{A}(\phi(s_M))$	auroc	$\mathcal{A}(\phi(s_M))$	auroc	$\mathcal{A}(\phi(s_M))$	auroc	$\mathcal{A}(\phi(s_M))$
Conf	64.88	-	64.61	-	69.23	-	69.41	-	67.16	-	66.35	-
Conf (**SCAO**)	72.13	-	72.01	-	**75.51**	-	73.64	-	72.78	-	72.46	-
Probe$_{dnn}$	77.09	8.54	75.70	11.64	72.79	11.17	76.43	10.68	75.54	10.19	71.03	12.26
Conf + Probe	79.10	10.55	77.54	13.48	70.77	9.15	**78.93**	13.18	**77.68**	13.33	70.80	12.03
Conf + Probe (**SCAO**)	**79.41**	10.86	**77.89**	13.83	74.89	13.27	78.86	13.11	77.22	12.87	**72.89**	14.12

(a) HotpotQA

	8B				70B			
	original		+ type		original		+ type	
	auroc	$\mathcal{A}(\phi(s_M))$	auroc	$\mathcal{A}(\phi(s_M))$	auroc	$\mathcal{A}(\phi(s_M))$	auroc	$\mathcal{A}(\phi(s_M))$
Conf	74.88	-	68.82	-	73.33	-	72.87	-
Conf (**SCAO**)	77.70	-	73.81	-	74.13	-	73.42	-
Probe$_{dnn}$	80.58	12.21	73.17	17.80	74.41	10.63	69.42	14.86
Conf + Probe	81.08	12.71	73.87	18.50	**77.33**	13.55	73.06	18.50
Conf + Probe (**SCAO**)	**83.39**	15.02	**75.51**	20.14	77.28	13.50	**73.52**	18.96

(b) ParaRel

	8B				70B			
	original		+ domain		original		+ domain	
	auroc	$\mathcal{A}(\phi(s_M))$	auroc	$\mathcal{A}(\phi(s_M))$	auroc	$\mathcal{A}(\phi(s_M))$	auroc	$\mathcal{A}(\phi(s_M))$
Conf	71.03	-	59.51	-	70.87	-	59.92	
Conf (**SCAO**)	69.23	-	73.12	-	73.45	-	**74.51**	
Probe$_{dnn}$	88.76	6.15	70.34	12.79	89.88	2.90	68.66	15.37
Conf + Probe	88.78	6.17	73.08	15.53	**90.01**	3.03	68.86	15.57
Conf + Probe (**SCAO**)	**88.95**	6.34	**76.09**	18.54	89.82	2.84	70.84	17.55

(c) Explain

	8B				70B			
	original		+ domain		original		+ domain	
	auroc	$\mathcal{A}(\phi(s_M))$	auroc	$\mathcal{A}(\phi(s_M))$	auroc	$\mathcal{A}(\phi(s_M))$	auroc	$\mathcal{A}(\phi(s_M))$
Conf	49.45	-	50.57	-	48.96	-	46.81	-
Conf (**SCAO**)	62.90	-	62.28	-	57.21	-	59.54	-
Probe$_{dnn}$	84.68	15.28	68.55	6.66	83.69	16.00	66.58	8.87
Conf + Probe	84.89	16.49	69.15	7.26	83.68	15.99	66.78	9.07
Conf + Probe (**SCAO**)	**85.42**	17.02	**70.04**	8.15	**84.94**	17.25	**68.67**	10.96

to the datasets from §4.3 that support refinement (Mintaka, HotpotQA, ParaRel, Explain). We also report the gap between the metric performance and AQE, denoted as $\mathcal{A}(\phi(s_M))$. However, we do not report the $\mathcal{A}(\phi(s_M))$ for the methods that use only the confidence score (e.g., *Conf*), because they lack question-side information, making it impossible to compute s_Q. In Table 11, *original* refers to the unrefined version of dataset, while + *type* and + *domain* indicate versions refined based on question type and domain, respectively. Detailed descriptions are in §D.

Results We observe the following points. (1) On the datasets of *original*, the performance shows very promising records of around AUROC 0.80, the highest among all versions. However, the $\mathcal{A}(\phi(s_M))$ is the smallest, suggesting that the performance is largely attributed to question-side shortcuts. (2) In refined versions (+ *type,* + *domain*), the performance measures sharply drop. For example, the AUROC for HotpotQA drops from 80.58 to 73.17 after refinement. This again demonstrates that the performance reported in previous works was largely driven by shortcuts.

(3) However, the AQE gap is larger in the refined datasets, indicating that when question-side effects are reduced, the use of model-side information becomes more activated. (4) Methods that rely solely on the confidence score (*Conf(SCAO)*) perform poorly on the original datasets, but show smaller performance variation across different data versions. And in some refined settings, it even outperforms other baselines. It is somewhat counter-intuitive that *Conf(SCAO)* outperforms hidden-state based methods, though it is provided significantly smaller amount of information. This suggests that the confidence score with SCAO captures a substantial amount of s_M, contributing to its strong generalization performance. (5) The *Conf + Probe (SCAO)* shows the largest $\mathcal{A}(\phi(s_M))$ across all refined versions of the datasets, suggesting a stable and effective direction for achieving self-awareness. (6) These trends remain consistent across models of different sizes and also hold under the accuracy metric §E.4.

(7) The $\mathcal{A}(\phi(s_M))$ is very low in OOD setting of Explain, which reveals that the hidden-state-based approach has limited generalization in long-form question answering settings. such as Explain. This contradicts the reports from previous works (Snyder et al., 2024; Chen et al., 2024).

6 Conclusion

In this work, we argued that hallucination prediction performance is often inflated by question-side shortcuts rather than reflecting genuine self-awareness. To disentangle these effects, we introduced Approximate Question-side Effect (AQE) and showed that many existing benchmarks exhibit strong question-aware signals, limiting the generalizability of prior results. Our analysis demonstrates that, once such effects are controlled, the contribution of true model-side awareness is substantially smaller than previously reported. Limitations are discussed in §A.

References

Amos Azaria and Tom Mitchell. 2023. The internal state of an llm knows when it's lying. *arXiv preprint arXiv:2304.13734*.

Jeffrey R Binder. 2015. The wernicke area: Modern evidence and a reinterpretation. *Neurology*, 85(24):2170–2175.

Jerrod Brown, D Huntley, S Morgan, KD Dodson, and J Cich. 2017. Confabulation: A guide for mental health professionals. *Int J Neurol Neurother*, 4:070.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. *Preprint*, arXiv:2005.14165.

Patrice Béchard and Orlando Marquez Ayala. 2024. Reducing hallucination in structured outputs via retrieval-augmented generation. *Preprint*, arXiv:2404.08189.

Chao Chen, Kai Liu, Ze Chen, Yi Gu, Yue Wu, Mingyuan Tao, Zhihang Fu, and Jieping Ye. 2024. Inside: Llms' internal states retain the power of hallucination detection. *Preprint*, arXiv:2402.03744.

Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2024. The faiss library. *arXiv preprint arXiv:2401.08281*.

Yanai Elazar, Nora Kassner, Shauli Ravfogel, Abhilasha Ravichander, Eduard Hovy, Hinrich Schütze, and Yoav Goldberg. 2021. Measuring and improving consistency in pretrained language models. *Preprint*, arXiv:2102.01017.

Ekaterina Fadeeva, Aleksandr Rubashevskii, Artem Shelmanov, Sergey Petrakov, Haonan Li, Hamdy Mubarak, Evgenii Tsymbalov, Gleb Kuzmin, Alexander Panchenko, Timothy Baldwin, and 1 others. 2024. Fact-checking the output of large language models via token-level uncertainty quantification. *arXiv preprint arXiv:2403.04696*.

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. 2019. Eli5: Long form question answering. *arXiv preprint arXiv:1907.09190*.

Daniel Fryer, Inga Strümke, and Hien Nguyen. 2021. Shapley values for feature selection: The good, the bad, and the axioms. *Ieee Access*, 9:144352–144360.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *Preprint*, arXiv:2009.03300.

Metehan Irak, Can Soylu, Gözem Turan, and Dicle Çapan. 2019. Neurobiological basis of feeling of knowing in episodic memory. *Cognitive Neurodynamics*, 13:239–256.

Raghav Jain, Daivik Sojitra, Arkadeep Acharya, Sriparna Saha, Adam Jatowt, and Sandipan Dandapat. 2023. Do language models have a common sense regarding time? revisiting temporal commonsense reasoning in the era of large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6750–6774, Singapore. Association for Computational Linguistics.

Daniel Kahneman. 2011. Thinking, fast and slow. *Farrar, Straus and Giroux*.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906*.

Asher Koriat. 1993. How do we know that we know? the accessibility model of the feeling of knowing. *Psychological review*, 100(4):609.

Junyi Li, Xiaoxue Cheng, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2023. Halueval: A large-scale hallucination evaluation benchmark for large language models. *Preprint*, arXiv:2305.11747.

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.

Xiang Li. 2023. Teach large language models the concept of meta-cognition to reduce hallucination text generation.

Yuan Li, Yue Huang, Hongyi Wang, Xiangliang Zhang, James Zou, and Lichao Sun. 2024b. Quantifying ai psychology: A psychometrics benchmark for large language models. *arXiv preprint arXiv:2406.17675*.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Truthfulqa: Measuring how models mimic human falsehoods. *Preprint*, arXiv:2109.07958.

Weitang Liu, Xiaoyun Wang, John D. Owens, and Yixuan Li. 2021. Energy-based out-of-distribution detection. *Preprint*, arXiv:2010.03759.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: Nlg evaluation using gpt-4 with better human alignment. *Preprint*, arXiv:2303.16634.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2022. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. *arXiv preprint arXiv:2212.10511*.

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

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. *Preprint*, arXiv:2305.14251.

Alain Morin. 2011. Self-awareness part 1: Definition, measures, effects, functions, and antecedents. *Social and personality psychology compass*, 5(10):807–823.

OpenAI. 2024. "hello gpt-4". 2(5).

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *Preprint*, arXiv:1606.05250.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *Preprint*, arXiv:1908.10084.

Jie Ren, Jiaming Luo, Yao Zhao, Kundan Krishna, Mohammad Saleh, Balaji Lakshminarayanan, and Peter J. Liu. 2023. Out-of-distribution detection and selective generation for conditional language models. *Preprint*, arXiv:2209.15558.

Armin Schnider. 2001. Spontaneous confabulation, reality monitoring, and the limbic system—a review. *Brain Research Reviews*, 36(2-3):150–160.

Priyanka Sen, Alham Fikri Aji, and Amir Saffari. 2022. Mintaka: A complex, natural, and multilingual dataset for end-to-end question answering. *Preprint*, arXiv:2210.01613.

Ben Snyder, Marius Moisescu, and Muhammad Bilal Zafar. 2024. On early detection of hallucinations in factual question answering. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2721–2732.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. Fever: a large-scale dataset for fact extraction and verification. *arXiv preprint arXiv:1803.05355*.

SM Tonmoy, SM Zaman, Vinija Jain, Anku Rani, Vipula Rawte, Aman Chadha, and Amitava Das. 2024. A comprehensive survey of hallucination mitigation techniques in large language models. *arXiv preprint arXiv:2401.01313*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. Attention is all you need. *Preprint*, arXiv:1706.03762.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. *Preprint*, arXiv:1809.09600.

Wenpeng Yin, Mo Yu, Bing Xiang, Bowen Zhou, and Hinrich Schütze. 2016. Simple question answering by attentive convolutional neural network. *Preprint*, arXiv:1606.03391.

Zhangyue Yin, Qiushi Sun, Qipeng Guo, Jiawen Wu, Xipeng Qiu, and Xuanjing Huang. 2023. Do large language models know what they don't know? *Preprint*, arXiv:2305.18153.

Hanning Zhang, Shizhe Diao, Yong Lin, Yi R. Fung, Qing Lian, Xingyao Wang, Yangyi Chen, Heng Ji, and Tong Zhang. 2024. R-tuning: Instructing large language models to say 'i don't know'. *Preprint*, arXiv:2311.09677.

A Limitation

Scope Narrowed to System 1 As discussed in the §2, our study focuses on System 1 (fast, automatic processing) rather than System 2. The knowledge recall setting we consider largely involves rapidly retrieving stored information in response to a prompt; accordingly, our analyses and conclusions should be interpreted primarily as describing phenomena that arise in retrieval/recall-driven, short-horizon judgments and answer generation.

In contrast, tasks dominated by System 2 (slow, deliberative processing)—such as multi-step reasoning, planning, long-form generation, and explicit verification—may exhibit qualitatively different sources and signals of failure. Therefore, the indicators and observations proposed in this work should not be assumed to directly generalize to System 2–heavy settings.

That said, this scope choice does not diminish the significance of our contribution. Because System 1 and System 2 reflect fundamentally different computational regimes and error profiles, concepts like self-awareness and hallucination detection are likely to carry distinct meanings and objectives in each regime. Our work provides a foundation for more precise definition, measurement, and analysis of these concepts in System 1 settings, and clarifies what may need to be preserved versus redesigned when extending to System 2 scenarios.

Devising robust methods How to design a more generalizable methodology remains an open question for future work. Although we propose SCAO as a methodology, it still exhibits a very high AQE (i.e., low genuine awareness) in long-form question answering. This indicates that hallucination prediction results for long-form QA should be interpreted with great caution when they are based on existing confidence- or hidden-state–based methods.

Moreover, this suggests that long-form answering involves more complex functions beyond simple knowledge recall, and therefore requires hallucination prediction approaches that go beyond solely leveraging a model's internal states.

B Other question-side shortcuts

Broken question The most commonly observed problem across all datasets is insufficient annotations of question and answer pairs. This occurs when questions and answers follow a one-to-many relationship, but the annotations fail to cover them.

For example, the ParaRel consists of question-answer pairs such as: "Q: What field does Daniel Bernoulli work in? A: physics". Although Daniel Bernoulli also worked in the fields of mathematics and medicine (one-to-many relation), only one label is provided, failing to cover all possible correct answers. If an LLM responds with a correct answer but different from the given label (e.g., mathematics), it would still be classified as hallucinated (i.e., k is annotated $False$), which is incorrect. If ϕ is trained to predict k using such $\{s, k\}$ pairs, it will likely become biased toward predicting $\hat{k} = False$ whenever the domain of question is "field of work". This means ϕ learns a domain classification task, not self-awareness. While this may improve prediction performance within the dataset, it lacks generalizability. It is because its performance is likely to drop only if the quality of the dataset improves, or if questions are given from unseen domains. As a study of Zhang et al. (2024) reports hallucination performance on this dataset, this shortcut might have influenced the reported scores. This issue is also found in other datasets such as SimpleQuestions, which includes instances like "Q: What is a TV action show? A: Genji Tsushin Agedama", that also fail to cover various possible answers.

The broken question problem is addressed when a question includes detailed constraints that restrict the one-to-many mapping between the question and possible answers. For example, in Mintaka, detailed constraints are added to the questions to ensure a one-to-one mapping (e.g., "Who was the director of The Goodfellas and attended school at New York University's School of Film?").

Domain Classifying the domain of a question (e.g., science, society) alone can provide a rough guess of k. For example, suppose a model that is extensively trained on science data but is unfamiliar with society or history domains. Since the model is likely to make more hallucination when given questions in the history domain. In this way, k can be biased toward $True$ or $False$ by domains, as shown in Figure 2. In such cases, the task of predicting k with ϕ is more a domain classification, which is question-aware and not model-aware.

When we consider a case of a human, it becomes clear that such shortcuts have limited effectiveness and are far from self-aware. For example, suppose this model is not familiar with the history domain but still has knowledge about Abraham Lincoln. If this model were human, he could easily "feel" his inner state and recognize that he possesses knowledge about Lincoln, despite not being familiar with other historical issues. However, if the model were an LLM and ϕ is trained to exploit only question-side shortcuts, ϕ would tell the model does not know about Lincoln, though the model actually possesses the knowledge of Lincoln. Therefore, while the domain information of a question can provide a naive approximation of k, relying on it imposes limitations on precision. This again highlights that utilizing self-awareness is the ultimate direction in precise hallucination prediction.

Table 5: The portion of correct answer for each question type, by `LLaMA-3-8B`. The rate for binary type is in **bold**.

(a) HotpotQA

Type	$p(k = True)$
Bridge	0.6828
Comparison	**0.8477**

(b) Mintaka

Type	$p(k = True)$
Entity	0.5508
numerical	0.4011
date	0.5968
string	0.6315
boolean	**0.7283**

Question type Question type (e.g., short-answer, multiple-choice) also provides a strong hint for predicting k. The average probability of $k = True$ (denoted as $p(k = True)$) for binary-choice questions is at least 0.5 for random choice, significantly higher than for open-ended questions with 0 for random choice, as described in Table 5. In such a case, ϕ may learn a shortcut where it automatically predicts $k = True$ whenever it detects a binary-choice question. HotpotQA, HaluEval, and Mintaka contain such binary-choice questions.

There are various other question-side shortcuts, which are described in the §B. These shortcuts can be identified by considering various scenarios in which they may act as shortcuts, and it is likely that some shortcuts remain undiscovered, as it is very subtle to determine. Therefore, manually identifying and removing them from datasets is nontrivial. That is why we introduce AQE in the next section, a method for approximately assessing the total effect of question-side shortcuts without human investigation.

Answerability Some questions are inherently unjudgeable in terms of correct or incorrect. This includes preference-based questions, hypothetical scenarios, and philosophical inquiries. For such questions, any answer could be considered correct or hallucinated, depending on the perspective of the evaluator. For example, if an LLM is asked "What

color do you like?" and responds "Blue," the correctness of this answer depends on the interpretation: 1) If correctness is judged based on contextual appropriateness, the answer is valid. 2) If correctness is judged based on the idea that LLMs cannot have personal preferences, the answer could be labeled as hallucinated. In this case, ϕ can exploit this pattern by simply identifying unanswerable questions and assigning a fixed label (either $k = 0$ or $k = 1$) across all such cases. This makes the hallucination detection process dependent on question-awareness, rather than assessing self-awareness.

The dataset SelfAware collects only unanswerable questions and categorizes them into different types (e.g., no scientific consensus, imagination, completely subjective, too many variables, philosophical). However, distinguishing these does not require self-awareness. Instead, it is primarily an act of reading comprehension, where the model identifies the nature of the question rather than assessing its own knowledge state.

Time-sensitive question Time-sensitive questions are inherently difficult for LLMs to answer accurately, as LLMs lack a robust understanding of time (Jain et al., 2023). As a result, questions involving temporal information will be biased toward hallucinated responses. Datasets such as HaluEval, Mintaka, and TruthfulQA include such questions (e.g., "How old is Barack Obama?", "When did the most recent pandemic occur?").

Complexity Complexity awareness is another question-dependent approach to estimating k. This aligns with the case of "too many variables" in answerability awareness, where, if a question is too difficult, the model is more likely to fail, making it advantageous to predict "unknown" by default. However, the notion of complexity is relative. If a model has extensive knowledge in a certain domain, it may still answer correctly even if the complexity is high. These attributes may connect complexity with category awareness. Additionally, our analysis suggests that questions within a single dataset tend to have similar levels of complexity. Therefore, distinguishing k based on complexity within a dataset is expected to be relatively rare in the general experimental setup.

C Details on SCAO

In this section, we propose Semantic Compression through Answering in One word (SCAO), a hallucination prediction method designed to maximize the utilization of s_M. While the ϕ predicts k from input s, which consists of both s_M and s_Q, ϕ tends to depend more on s_Q which offers an easier shortcut, as discussed in §4.1. To prevent this problem, we need a method that strengthens the preference of ϕ for s_M.

To address this, we focus on the confidence score as a source of s. In §4.2 we noted that the confidence score is a 1-dimensional scalar where information is extremely saturated, which in turn makes it unlikely to carry high-level information of the question (s_Q). This makes the confidence score closer to s_M ($s \approx s_M$). Therefore, using the confidence score alone or aggregating it with the hidden state may increase the model's dependency on s_M.

However, as the confidence score is highly saturated and carries limited information, an approach is needed to amplify the information contained in it. SCAO is a method designed to more effectively express confidence in knowledge that can be represented as entities. The approach is straightforward: we insert a system instruction before the question, prompting the model to "you must answer in one word". The rationale behind why this technique can improve the use of s_M is as follows.

Causal LLM is an entity retriever when compressed The confidence score is calculated by the maximum inner product score between the last hidden state of θ and the token embedding vectors (i.e., decoder head). We analyze that this structure is analogous to the maximum inner product search (MIPS) used in dense retrieval (Karpukhin et al., 2020) (Figure 3). And we focus on the calibration function of similarity scores in dense retrieval. Previous research on dense retriever systems such as Faiss (Douze et al., 2024) leverages this calibration through a *range search*, which finds all the document vectors that are within some distance threshold.

This concept can be interpreted in reverse that we can evaluate whether knowledge is within the vector DB, with a fixed confidence threshold. For example, suppose that a vector database contains documents about the biography of Newton but contains rare data about the biography of Lincoln. Querying "Give me an explanation on Lincoln" may result in few documents with confidence scores over the threshold. In contrast, querying "Give me an explanation on Newton" would likely retrieve a greater number of documents exceeding the threshold, reflecting a stronger alignment between the query

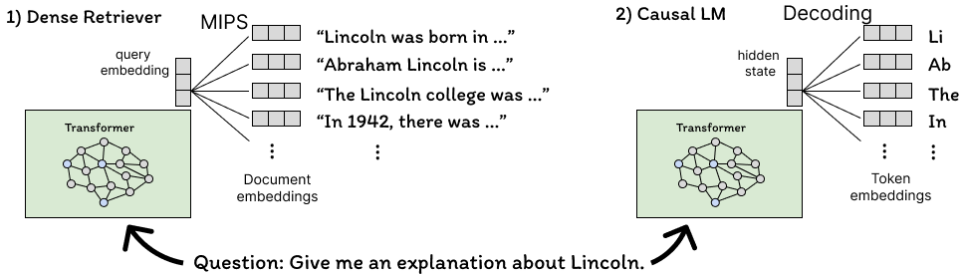

Figure 3: Structural analogy between 1) dense retriever and 2) causal LM.

and the knowledge contained in the database.

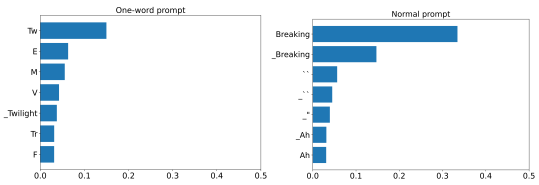

Figure 4: Y-axis is the top-7 candidates of the first token of the answer to the question "Please give me an explanation about **Breaking Dawn**". The X-axis is the probability for each candidate. **Left** is for one-word prompt, and the **Right** is for normal prompt.

However, a causal LM not only performs entity retrieval but also generates full sentences by connecting these words. Such considerations of grammatical words and sentence structure may act as noise for a calibration signal. Therefore, by minimizing the consideration of sentence structure in the model, the LLM will become more analogous to an entity retriever. If LLM becomes more analogous to an entity retriever, its behavior will become more similar to the calibration properties in dense retrieval. A straightforward way to minimize the consideration of grammatical context in the model is to instruct it to "you must answer in one word", under the assumption that the model is well-trained to follow instructions.

In Figure 4, we show the confidence pattern at the first token of the answer, with and without SCAO when the model is provided a question "Please give me an explanation about Breaking Dawn". With the one-word prompt (Left), the model appears to attempt to retrieve knowledge related to "Twilight," which is the series name of Breaking Dawn. In contrast, with the normal prompt (Right), the model tends to repeat the question entity, "Breaking Down". Since it chooses the easier path, the overall probabilities are higher. Further analysis of the confidence pattern is in §C.2. In §5, we empirically show that applying SCAO leads

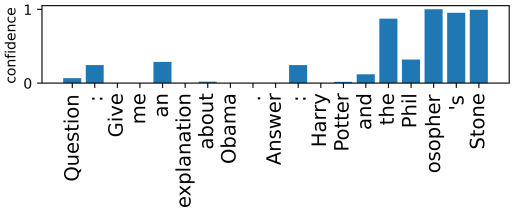

Figure 5: Probability pattern of the hallucinated answer, by LLaMA3-8B. Each bar stands for the probability (0,1) of the corresponding token.

to improved hallucination prediction performance, especially more in settings with lower AQE dataset, where the use of model-side information becomes more critical.

C.1 Efficiency of first token as a discriminator

Previous works on confidence-based hallucination detection research mostly utilize the confidence score of all tokens in the answer sentences, with normalization such as averaging (Chen et al., 2024). Utilizing more information is ultimately more advantageous; however, it also has several drawbacks. We observe a pattern that as the entity name length increases, the average confidence tends to rise. For example, Figure 5 depicts the confidence pattern of the hallucinated question-answer pair "Question: Give me an explanation about Obama. Answer: Harry Potter and the Philosopher's Stone".

Up to the token "Harry Potter", the confidence is near zero since it conflicts with the question. However, from a "philosopher", confidence increases to a near maximum, as the previous context of "Harry Potter" supports it strongly. Thus, the average confidence tends to increase regardless of whether it makes sense, when the entity name gets longer or the sentence contains more grammatical elements. This observation is supported by the analysis in Figure 6 (Left), which shows that the correlation between the mean confidence and the k tends to decrease as the token increases.

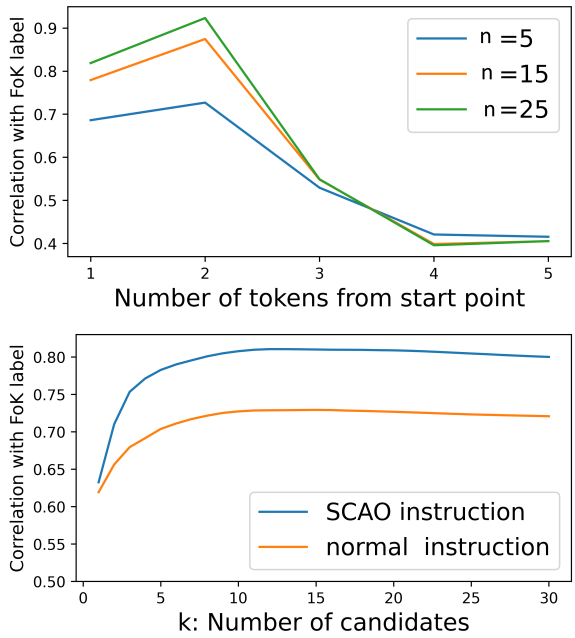

Figure 6: Y-axis is a correlation between the mean confidence and the k. The X-axis of each figure stands for (**Left**) the number of tokens from the start point of the answer, and (**Right**) the number of candidates used to calculate the mean. The LLaMA3-8B and (s, k) datasets from Mintaka are utilized.

We also observe that averaging the confidence scores across top-n vocabulary candidates, rather than just the top-1, shows a stronger correlation with the label k, particularly peaking around n=15 (Figure 6 (Right)). This suggests that incorporating more samples of distance provides more information about the relationship between the output vector and the token space.

C.2 Confidence pattern of SCAO

Table 6: The number and portion of each case, when questions from the test set (total 2152) of Explain are asked to the `LLaMA-3-8B-Instruct` model using various prompts. The columns represent each prompt style. In the rows, "repeating subject" refers to cases where the top-1 candidate for the first token of the answer is a component of the queried subject entity. "The" refers to cases where the top-1 token is "the."

	one-word prompt	normal prompt
repeating subject	1819 (84.5%)	261 (12.1%)
"the"	383 (17.7%)	5 (0.2%)

In Figure 4, we show how the model reacts at the first token of the answer in both one-word prompts and the normal prompt.

First, in non-compressed cases (queried with a normal prompt), the following patterns are frequently observed: (1) The response often starts by repeating the entity name mentioned in the query. (2) The response begins with grammatical function words such as "The" or "A". In other words, the model tends to take the easy path. As a result, the probability of the initial token is generally inflated, regardless of whether the model truly knows the subject.

On the other hand, when prompted to answer with a one-word response, the first token often corresponds to the initial token of a word encapsulating the entity's characteristics. For example, in response to the question "Please give me an explanation about 'Breaking Dawn'.", the first candidate token was "Tw" (the first token of "Twilight"). In other words, with one-word prompting, the model shows a stronger tendency to retrieve its own knowledge related to the entity. This trend is also reflected statistically. Among the 2152 test samples in the Explain dataset, the case where the top-1 candidate of the first token of the response is a component of the entity is 84.5% for normal prompting, significantly outpacing the 12.1% for one-word prompting. Similarly, the first token being "the" occurred in 17.8% of normal prompting cases, compared to just 0.02% for one-word prompting. (Table 6)

C.3 Rationale on why confidence-based method is better in generalization

In §5, we observed that the confidence-based method (*SCAO*) outperforms the hidden-state-based method (probing) in the out-of-domain setting. This result is counter-intuitive, as confidence scores are highly saturated scalar values, whereas hidden states are high-dimensional vectors capable of carrying much richer information. We suggest the following rationale for this result, examining how the probing and SCAO learn to predict hallucinations

SCAO and probing are fundamentally similar. Probing directly utilizes the raw hidden state of θ, while SCAO focuses on the last hidden state of θ, which is projected onto the vocab embedding space.

Let us assume a knowledge space (denoted as S_k) (Figure 7), which represents the embedding of each knowledge in the θ. And we term the gray area in the S_k as a **boundary of knowing** of θ, which represents the area where $k = True$ (model possesses the knowledge). This space is hypothetical

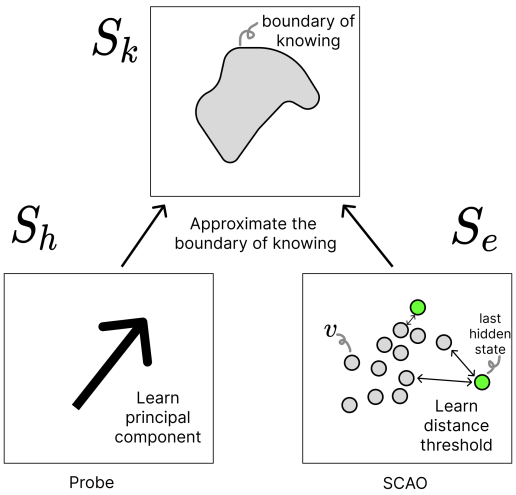

Figure 7: Illustration on two methods (probe, SCAO) approximating the boundary of knowing of θ. In S_e (lower right), the green balls are the last hidden state vector that is mapped to the vocab space. SCAO learns the threshold of distance between the hidden state and v to classify y of each ball.

and unknown but needs to be discovered to predict hallucination of θ. To approximate this, what we have at hand are (1) the 4096-dimensional (in the case of `LLaMA-3-8B-Instruct`) hidden states (denoted as S_h) and (2) a vocab embedding space (denoted as S_e) of the same dimension, with vocab embedding vectors (denoted as v) distributed across S_e.

In probing, a linear layer is trained to map S_h to S_k. The weight of the linear layer is supposed to be a direction vector that represents a principal component of the boundary of knowing. Thus, an inner product with this vector tells if the given hidden states match the direction. Since it utilizes all 4096 dimensions to describe S_k, it offers high informational resolution, leading to generally strong performance.

Conversely, SCAO assumes that S_e approximately aligns with S_k, which means the v (gray balls in Figure 7) aligns with the boundary of knowing (gray area in Figure 7). SCAO figures the shape of S_k by measuring the distance between the hidden state vector from the last layer (green balls in Figure 7) and embedding vectors v. These mechanisms yield the following properties: (1) SCAO leverages S_e and S_h, thus utilizing more information than probing, which only uses S_h. (2) However, this information is compressed into a single scalar value, distance, leading to lower information resolution, showing lower performance than the probe. 3) Despite the lower resolution, this simplification appears to enhance generalization. For instance, in out-of-domain scenarios, probing struggles with unfamiliar features in S_h, while SCAO effectively handles these novel features by employing its simplified distance-based measure.

Since probing and SCAO reflect slightly different aspects of S_k, combining these two methods in a feature fusion appears to provide an additional performance boost by leveraging their complementary strengths.

D Detail of datasets

D.1 Datasets and their refinement strategies

In this paragraph, we present details about the benchmark dataset for evaluating the hallucination prediction method: Mintaka (Sen et al., 2022), ParaRel (Elazar et al., 2021), HaluEval (Li et al., 2023), HotpotQA (Yang et al., 2018), and Explain. We also describe the refinement strategies applied to each dataset to reduce their AQE. "+ type" refers to refinements related to question types, while "+ domain" refers to refinements based on question domains, following Table 11

Mintaka Mintaka is a challenging multilingual QA dataset consisting of 20,000 question–answer pairs collected from MTurk contributors and annotated with corresponding Wikidata entities for both questions and answers. Mintaka includes five types of question–answer pairs (entity, boolean, numerical, date, and string) and eight categories (movies, music, sports, books, geography, politics, video games, and history). Among multiple languages, we only use English question-answer pairs.

(1) + type : Among five types, we exclude boolean and numerical questions, following the discussion in §4.1 and §B. (2) + domain : We randomly selected half of the domains (books, movies, music, sports) as the training and validation sets, and assigned the remaining domains to the test set.

ParaRel ParaRel is originally a dataset designed for masked language modeling, containing factual knowledge expressed through diverse prompt templates and relational types. We utilize the rearranged version by (Zhang et al., 2024). This version consists of 25,133 prompt-answer pairs across 31 domains. It is further divided into two parts: the first 15 domains are classified as in-domain data, and the remaining 16 domains are classified as out-of-domain.

(1) + domain: In the original setting, in-domain data was used as the test set. In the refined setting, the test set consists of out-of-domain data.

HotpotQA HotpotQA is a question-answering dataset where each instance consists of a question, label (types including entity, boolean, numerical), and reference documents. We utilize only the question and answer to fit the closed-book scenario. An example of the question is "What government position was held by the woman who portrayed Corliss Archer in the film Kiss and Tell?", paired with the label "Chief of Protocol". We use the development dataset as a test set, following (Zhang et al., 2024).

(1) + type: We exclude the "comparison" type, as it consists of yes/no questions or questions that require choosing between two given options.

Explain We present a benchmark **Explain** to evaluate a model's ability to provide a descriptive answer to an open-ended question. Explain is an extended and refined version of an open-ended long-form dataset in the well-known and verified work of FActScore (Min et al., 2023). In FActScore, a small dataset is devised to test the fact-checking pipeline for long-form QA. This dataset is created by appending prompts like "Tell me a bio of <entity>" to person names sourced from Wikipedia. However, its subjects are limited to only person names, and it includes only 500 entries.

To address this, we developed Explain. Explain covers more general categories such as people, history, buildings, culture, etc (the entities from Mintaka), with the dataset size expanded to about 15000 entries. The prompt is "Please give me an explanation about <entity>", which follows the concept of the dataset in FActScore. All entities used as objects in the Explain setting are sourced from entity-type questions in the Mintaka dataset.

(1) + domain: Since the entities in Explain are sourced from Mintaka, we adopt the same domain-splitting strategy as used in Mintaka.

D.2 Data statistics

We present data statistics of our main benchmarks, Mintaka and ParaRel. We also present examples of questions, categories, and statistics on Explain (Table 9, Table 7)

Table 7: #data in each benchmarks

	ParaRel	Mintaka	HaluEval	HotpotQA	Explain
Train	5575	7583	6000	8000	7583
Valid	5584	1075	2000	2000	1075
Test	13974	2152	2000	7405	2152

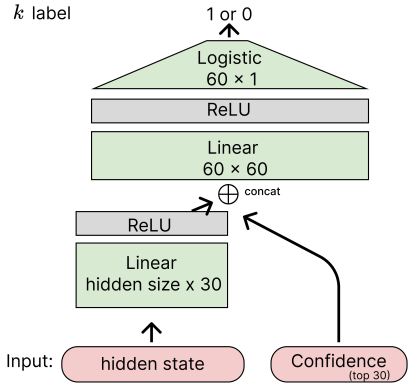

Figure 8: Structure of aggregation of hidden-state and confidence scores.

E Experimental detail

E.1 Detail on the three main approaches

In this section, we provide detailed descriptions of the three main approaches evaluated in §5. (1) confidence-based: We adopt a simplified method that takes the top-n softmax probabilities of the first answer token and applies a threshold. n and the threshold t are learnable ϕ.

For the threshold-based discrimination, we first use the mean value of top-n vocabulary (s_j for j_{th} token confidence in top-n candidates) and apply the threshold, as depicted in Equation 5. Here, the learnable parameter $\phi = \{t, n\}$ consists of a threshold (t) and the number of vocabulary candidates (n). During the training session, every possible pair of n and threshold (n is 1 to 30 in 30 steps, t is 0 to 0.1 in 3000 steps, total 90K $\{t, n\}$ pairs) are measured on the training dataset, and the pair with the highest accuracy is applied to the test session.

$$\phi(s) = \begin{cases} 1, & \text{if } \frac{1}{n}\sum_{j=1}^{n} s_j \geq t \\ 0, & \text{if } \frac{1}{n}\sum_{j=1}^{n} s_j < t \end{cases} \quad (5)$$

(2) hidden-state-based: We employ a 3-layer deep neural network (DNN) structure with dimensions $d \to 100 \to 30 \to 1$, where d is the hidden size. ReLU activation is applied between each layer. The objective function of DNN is binary cross entropy loss $L = -\frac{1}{N}\sum[y \cdot \log(\phi(s)) + (1-y) \cdot \log(1-\phi(s))]$. DNN ($\phi$) is trained on the dataset

Table 8: Examples of questions in Explain

Questions	Entity
Please give me an explanation about “A Game of Thrones”.	A Game of Thrones
Please give me an explanation about “Simone Biles”.	Simone Biles
Please give me an explanation about “Winston Churchill”.	Winston Churchill
Please give me an explanation about “Fyodor Dostoevsky”.	Fyodor Dostoevsky
Please give me an explanation about “District 12”.	District 12
Please give me an explanation about “The Battle of Gettysburg”.	The Battle of Gettysburg

Table 9: #Data for the entity domains in Explain

	Train	Dev	Test
Music	914	139	273
History	1059	149	296
Geography	1033	144	306
Politics	1036	143	300
Video games	1057	150	302
Movies	953	138	269
Books	1020	140	283
Sports	909	128	245

while θ is frozen. The choice of which layer's hidden state from θ to use is determined during the training phase, based on the one that achieves the highest validation accuracy. This approach extends the linear probing (Li et al., 2024a).

We analyze that DNN emulates the mechanism of the mean threshold approach. The weights of the first layer decide how many candidates to count in, corresponding to the function of n in the threshold-based approach. The second layer decides operations, such as mean or max pooling. DNN structure is a more suitable choice if feature fusion with other data is required.

(3) aggregation: We utilize the feature fusion of confidence and hidden state. That implies utilizing both top-30 confidence value and h_{th} hidden state from θ as inputs to DNN. This approach concatenates the confidence scores and hidden state into a single vector, which is then passed to a learnable ϕ (Snyder et al., 2024). We use a module with dimensions of $(d + n) \rightarrow 100 \rightarrow 30 \rightarrow 1$, where n is fixed to 30 (Figure 8).

E.2 Experiment pipeline

First the dataset is divided into D_{train}, D_{valid}, and D_{test}. We fit ϕ to D_{train}, while θ is frozen. The next step varies between the two types.

Learning-based The methods with hidden-state (*Probe* and *Probe + Conf*) should employ DNN architecture for ϕ, which need machine learning. In this case, ϕ is trained on the D_{train} with the objective of BCELoss. We train for 20 epochs to get ϕ. And from the final checkpoint, we choose the index of the hidden layer of θ with the best accuracy on the D_{valid}. Then we use this hidden layer index and ϕ to test on the D_{test}. We calculate two metrics of accuracy and AUROC. When training, the learning rate is 1e-3, and the optimizer is AdamW.

Threshold-based The method that solely relies on confidence score (*Conf*) uses a threshold for calibration. Here, learnable ϕ is the number of top candidate confidence scores (n) and the threshold (t). These two parameters are fitted in D_{train}, without evaluation on D_{valid}. We select the ϕ that achieves the highest accuracy by performing a grid search over t values in the range $[0, 1]$ with 1000 uniformly spaced points, and n in the range $[1, 30]$ with 3000 uniformly spaced points. And use this ϕ to test on the D_{test}. AUROC is measured only with n_ϕ, without t_ϕ.

E.3 AQE of larger model

In this section, we provide the AQE results of the larger model (`LLaMA-3-70B-Instruct`) on the refined dataset, corresponding to Table 10.

E.4 Experiment with accuracy

In this section, we present the performance and accuracy-based deltas of various hallucination prediction methods. The results exhibit trends similar to those observed when using AUROC as the evaluation metric.

-

Table 10: AQE score of dataset and `LLaMA3-70B-Instruct`. The version (original, type , domain) with the lowest AQE within each dataset is highlighted in **bold**.

	Mintaka			HotpotQA		ParaRel		Explain	
	original	+ type	+ type + domain	original	+ type	original	+ domain	original	+ domain
$p(k = True)$	62.17	57.28	57.42	33.81	26.64	51.71	53.01	55.71	54.35
$p(k = False)$	37.82	42.71	42.57	66.18	73.35	48.28	46.98	44.28	45.64
AQE_{acc}	65.52	62.15	**58.96**	**68.06**	71.44	76.68	**53.22**	61.96	**55.31**
AQE_{auc}	65.75	64.35	**58.77**	63.78	**54.56**	85.98	**53.29**	67.69	**57.71**

Table 11: Hallucination prediction performance (accuracy) of instruction-tuned 8B and 70B LLaMA models across multiple datasets.

(a) Mintaka

	8B						70B					
	original		+ type		+ type + domain		original		+ type		+ type + domain	
	auroc	$\mathcal{A}(\phi(s_M))$	auroc	$\mathcal{A}(\phi(s_M))$	auroc	$\mathcal{A}(\phi(s_M))$	auroc	$\mathcal{A}(\phi(s_M))$	auroc	$\mathcal{A}(\phi(s_M))$	auroc	$\mathcal{A}(\phi(s_M))$
Conf	62.75	-	60.71	-	61.54	-	68.39	-	65.53	-	64.65	-
Conf (**SCAO**)	67.35	-	65.78	-	66.75	-	70.53	-	68.15	-	66.98	-
Probe_{dnn}	70.55	7.05	68.87	9.06	66.68	7.64	73.32	7.8	69.73	7.58	66.12	7.16
Conf + Probe	71.38	7.88	69.98	10.17	65.12	6.08	**74.21**	8.69	**71.44**	9.29	65.99	7.03
Conf + Probe (**SCAO**)	**71.96**	8.46	**79.41**	10.68	**68.21**	9.17	74.14	8.62	71.35	9.20	**67.61**	8.65

(a) HotpotQA

	8B				70B			
	original		+ type		original		+ type	
	acc	$\mathcal{A}(\phi(s_M))$	acc	$\mathcal{A}(\phi(s_M))$	acc	$\mathcal{A}(\phi(s_M))$	acc	$\mathcal{A}(\phi(s_M))$
Conf	71.54	-	75.93	-	69.75	-	73.08	-
Conf (**SCAO**)	73.23	-	76.93	-	70.97	-	**75.97**	-
Probe_{dnn}	75.49	6.94	76.31	0.28	71.24	3.18	71.91	0.47
Conf + Probe	76.00	7.45	75.46	-0.56	72.50	4.44	74.02	2.58
Conf + Probe (**SCAO**)	**77.69**	9.14	**76.95**	0.92	**72.51**	4.45	74.90	3.46

(b) ParaRel

	8B				70B			
	original		+ domain		original		+ domain	
	acc	$\mathcal{A}(\phi(s_M))$	acc	$\mathcal{A}(\phi(s_M))$	acc	$\mathcal{A}(\phi(s_M))$	acc	$\mathcal{A}(\phi(s_M))$
Conf	67.58	-	62.63	-	65.75	-	57.29	-
Conf (**SCAO**)	66.67	-	67.88	-	66.83	-	**69.19**	-
Probe_{dnn}	80.52	7.26	66.82	11.73	**82.29**	5.61	65.24	12.02
Conf + Probe	80.64	7.38	68.65	13.56	82.18	5.50	64.46	11.24
Conf + Probe (**SCAO**)	**80.92**	7.66	**69.24**	14.15	81.84	5.16	66.36	13.14

(c) Explain

	8B				70B			
	original		+ domain		original		+ domain	
	acc	$\mathcal{A}(\phi(s_M))$	acc	$\mathcal{A}(\phi(s_M))$	acc	$\mathcal{A}(\phi(s_M))$	acc	$\mathcal{A}(\phi(s_M))$
Conf	57.66	-	60.29	-	55.71	-	53.31	-
Conf (**SCAO**)	59.71	-	59.57	-	60.99	-	60.78	-
Probe_{dnn}	78.92	13.27	63.31	2.1	76.28	14.32	62.65	7.34
Conf + Probe	79.12	13.47	64.78	3.57	76.00	14.04	63.10	7.79
Conf + Probe (**SCAO**)	**80.91**	15.26	**63.66**	2.45	**78.75**	16.79	**64.87**	9.56

F Instruction prompts

In this section, we compile the instructional prompts employed in our study. Terms marked with `underline` indicate placeholders that need to be filled with the corresponding content.

A. Normal instruction template

[Question]:{`question`} [Answer]:

B. SCAO instruction template

[Question]: {`question`} You must answer in only one word. [Answer]:

C. G-eval instruction template

[Instruction] The text enclosed within triple backticks (“‘) is an agent’s answer to a given question ([Question]).
Your task is to evaluate whether the agent’s response is factually correct.

1) Analyze and explain whether the answer contains any factual inaccuracies.
2) Then, classify the answer as either "True" (only factually correct contents) or "False" (containing any factually incorrect content).

“‘
[Question]:{`question`}
[Answer]:{`answer`}
”’

G Usage of AI assistants

In preparing this manuscript, we relied on AI-powered writing tools to refine sentence flow, fix grammatical mistakes, and improve readability. These assistants were used strictly for language polishing and played no role in shaping the technical content, research design, or experimental work. All scientific concepts, findings, and conclusions presented in this paper were fully developed and written by the researchers. The involvement of AI was limited to editorial support and did not influence the originality or intellectual contributions of the study.