Bold Claims or Self-Doubt? Factuality Hallucination Type Detection via Belief State

Anonymous ACL submission

Abstract

Large language models are prone to generating hallucination that deviates from factual information. Existing studies mainly focus on detecting the presence of hallucinations but lack a systematic classification approach, which hinders deeper exploration of their characteristics. To address this, we introduce the concept of belief state, which quantifies the model's confidence in its own responses. We define the belief state of the model based on self-consistency, leveraging answer repetition rates to label confident and uncertain states. Based on this, we categorize factuality hallucination into two types: Overconfident Hallucination and Unaware Hallucination. Furthermore, we propose **BAFH**, a factuality hallucination type detection method. By training a classifier on model's hidden states, we establish a link between hidden states and belief states, enabling efficient and automatic hallucination type detection. Experimental results demonstrate the effectiveness of BAFH and the differences between hallucination types.

1 Introduction

Large Language Models (LLMs) have demonstrated remarkable capabilities in various Natural Language Processing tasks (Achiam et al., 2023). However, there's a concerning trend where they exhibit an inclination to generate hallucination (Cohen et al., 2023; Ren et al., 2023; Kuhn et al., 2023), which makes it risky to deploy LLMs in practical scenarios. Consequently, accurately detecting and addressing hallucination has become a significant research challenge (Azaria and Mitchell, 2023).

Existing LLM hallucination detection methods mainly focus on identifying factual errors in LLM outputs, which are commonly referred to as factuality hallucination (Lin et al., 2022a; Li et al., 2023; Manakul et al., 2023). For instance, Chern et al. (2023) utilize external tools for evidence gathering to detect factual errors. If the model's output does not align with evidence, it is considered a potential hallucination (Manakul et al., 2023; Zhang et al., 2023a; Azaria and Mitchell, 2023). Another category of methods do not rely on external knowledge, but instead detect hallucination by estimating the uncertainty of model outputs (Varshney et al., 2023; Luo et al., 2023; Yao et al., 2024). For example, MIND (Su et al., 2024) leverages the hidden states of LLMs for real-time hallucination detection without requiring manual annotations.

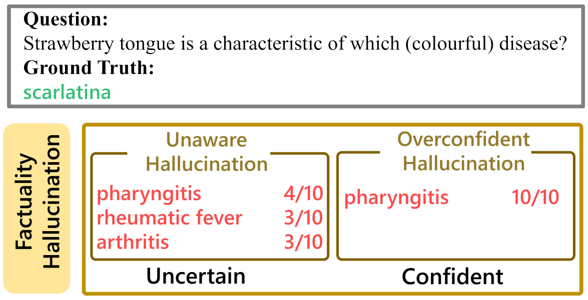


Figure 1: Our proposed two types of factuality hallucination. Red represents incorrect, and green represents correct.

Despite significant progress in factuality hallucination detection, existing work still has notable limitations. Current research primarily focuses on detecting the presence of factuality hallucination, with insufficient attention given to the detailed analysis of specific types of hallucination. A few studies (Huang et al., 2023a; Zhang et al., 2023b) that have attempted to classify hallucination typically base on semantic errors (e.g., factual or logical errors), but they lack a general classification framework and automated methods. These limitations constrain deeper understanding of the hallucination mechanisms in LLMs.

Therefore, we focus on factuality hallucination and pose the critical questions: "*Can factuality hallucination be categorized into distinct types? How can we effectively differentiate between these* *types of hallucination?*"

Manakul et al. (2023) point out that models exhibit varying degrees of uncertainty about their own answers. Inspired by this, we analyze and conclude that models also exhibit different levels of uncertainty about the hallucinations they generate. We describe this uncertainty as the model's belief state and propose a belief-state-based classification paradigm for factuality hallucination, as illustrated in Figure 1. We define belief states by measuring the consistency across different responses. As analyzed and discussed in Section 3, we categorize them into two types: confident state and uncertain state. Hallucinations generated by the model in confident belief state are referred to as **Overconfident Hallucinations**, while those generated in uncertain belief state are termed **Unaware Hallucinations**.

Given these considerations, we developed **B**elief-State-**A**ware **F**actuality **H**allucination Type Detection (BAFH) method, a lightweight framework that integrates with Transformer-based LLMs. BAFH leverages hidden states to determine belief states and classify hallucination types. In summary, our contributions are as follows:

• We analyzed the distribution of model responses based on self-consistency and proposed a new classification framework for factuality hallucination, which divides hallucinations into two types based on the model's belief states.

• We propose BAFH, which leverages the hidden states of large language models to analyze belief states and detect different types of hallucination.

• Experiment results demonstrate that BAFH achieves high accuracy on multiple datasets, while maintaining stability under various hyperparameter settings. These findings highlight the rationality of classification method we introduce and underscore the necessity of classifying hallucinations.

2 Related Work

Factuality Hallucination Detection Existing LLM hallucination detection methods primarily focus on factuality hallucination (Lin et al., 2022a; Li et al., 2023; Manakul et al., 2023) and can be divided into evidence-based and uncertainty-based methods. Evidence-based methods utilize external knowledge sources to verify model outputs. For instance, FACTSCORE (Min et al., 2023) determines the veracity of long-format text by decomposing LLM-generated content into atomic facts and calculating the percentage of atomic facts supported by reliable sources. Uncertainty-based methods detect hallucination by analyzing the model's hidden states or behavior (Slobodkin et al., 2023). For example, SAPLMA (Ji et al., 2024) and MIND (Su et al., 2024) use hidden states to construct classifiers, while Selfcheckgpt (Manakul et al., 2023) detects hallucinations by comparing the consistency of multiple responses. Although these methods have shown significant efficacy, they cannot distinguish between specific types of hallucinations or deeply explore the relationship between accuracy and the model's confidence in its answers.

Hallucination Classification In early studies, hallucinations are broadly categorized into intrinsic and extrinsic hallucinations based on whether the correctness of the output could be verified against the source content (Li et al., 2022; Huang et al., 2023b; Ji et al., 2023). Recent research has expanded these classifications to encompass hallucinations in broader contexts. For example, considering the user-centered interaction emphasized by LLMs, Huang et al. (2023a) classify hallucination into factuality and faithful hallucination. Faithful hallucination reflects the logical consistency within the generated content (Zhang et al., 2023b). Factuality hallucination refers to outputs containing factual inaccuracies that can be verified against reliable sources. While existing frameworks provide valuable insights (Zhang et al., 2023b; Huang et al., 2023a), their classification basis is often limited to task-specific or semantic levels, making them inadequate for comprehensively describing the complex generative behaviors of LLMs. To this end, we propose a new classification criteria and a detection method for factuality hallucination types and conduct comparative analysis of the characteristics of different hallucination types.

3 Motivation

In this section, we analyze the repetition count of model responses. Manakul et al. (2023) point out that the self-consistency of model responses reflects the model's uncertainty, which we refer to as belief states. Specifically, we define belief states as the model's internal confidence level in its generated responses, which can be inferred indirectly through the consistency of repeated outputs. To quantify belief states, we prompt the model to generate ten responses for each question and extracted the answers (as described in Section 4). Then we record the repetition count of the most frequent

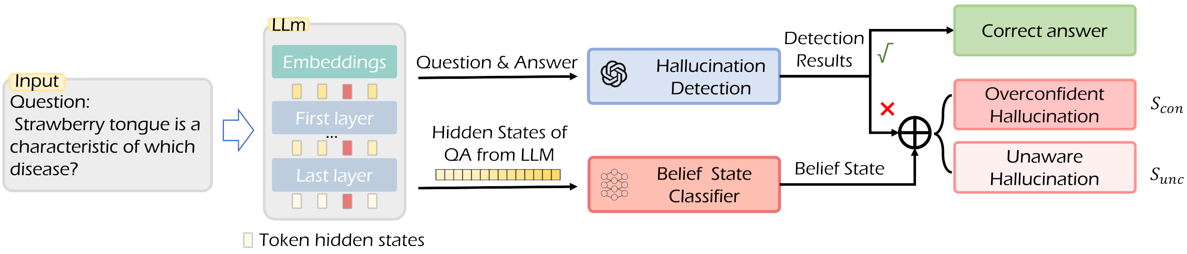


Figure 2: The overall process of BAFH

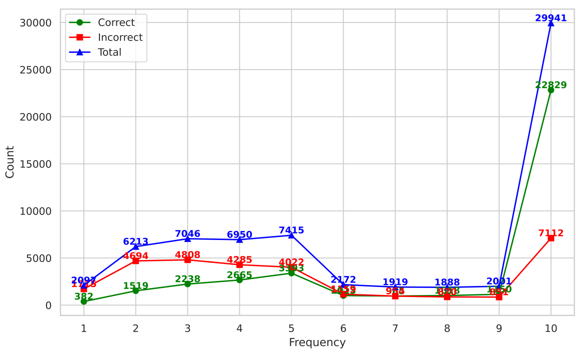


Figure 3: Statistical Analysis of Model Response Consistency for Gemma-2-9b-it on NQOPEN.

response along with its correctness.

Figure 3 shows that as repetition count increases, correct responses become more frequent, indicating that a higher repetition count is generally linked to greater confidence and accuracy. Moreover, while most factual errors have a low repetition count, some still occur with high repetition. This suggests that LLMs may retain high confidence even when generating hallucinations, implying that not all hallucinations stem from uncertainty.

Notably, response repetition counts exhibit an uneven distribution, with higher counts (e.g., 10) and lower counts (e.g., 1–5) being more common, while intermediate counts (e.g., 6–8) are relatively rare. This observation suggests a potential bimodal tendency in the behavior of LLMs (More details are provided in Appendix A.1). Based on these observations, we hypothesize that this distribution may reflect a clustering of the model's belief state around two primary modes, which we refer to as **confident state** and **uncertain state**. Correspondingly, we categorize the hallucinations arising from these states as Overconfident Hallucinations and Unaware Hallucinations. Section 4 presents our hallucination type detection method.

4 Method

4.1 Overview

We define the task of detecting factuality hallucination types as a binary classification problem: determining whether a hallucination produced by a model is an Overconfident Hallucination or an Unaware Hallucination. To this end, we propose BAFH consisting of two core modules: a belief state classifier and an evidence-based hallucination detection module, as illustrated in Figure 2.

Given a question, BAFH extracts the hidden states from the LLM during the answer generation process. The hallucination detection module employs the method from Li et al. (2023), utilizing ChatGPT to assess the correctness of the model's response. The belief state classifier employs a feedforward neural network, which takes the hidden states from the generation process as input and outputs the model's belief state (confident or uncertain). BAFH then combines the hallucination detection result and the belief state to categorize the hallucination as either an Overconfident Hallucination or an Unaware Hallucination.

4.2 Belief State Classifier

To obtain the model's belief state, we train a classifier based on a feedforward neural network. As shown in Figure 4, we first evaluate the model's belief state and construct a training set by associating belief state labels with hidden states obtained during answer generation. This training set is then used to train a model-specific belief state classifier.

Belief State Evaluation Evaluating the belief state is a crucial step in constructing the training dataset. In this paper, we define the belief state as the model's confidence level in its own answer, independent of the question's answerability or the correctness of the response. We categorize the model's belief state regarding its own answer into two types: confident state and uncertain state.

Inspired by Kadavath et al. (2022) and Lin et al.

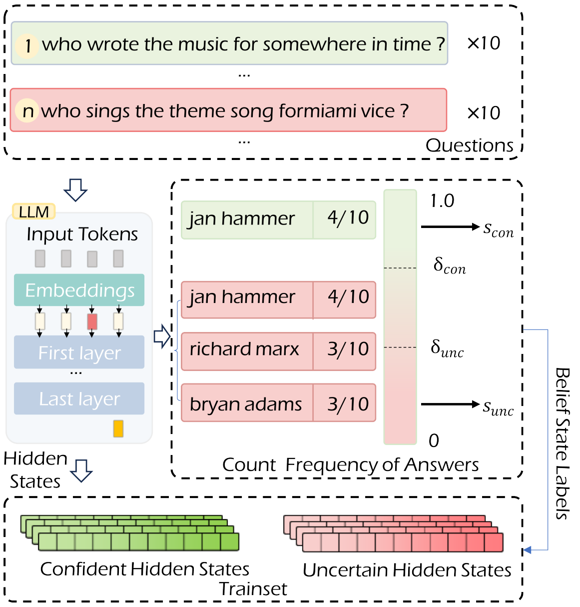


Figure 4: Constructing the Belief State Training Dataset

(2022b), we determine the belief state by assessing the self-consistency of the model's answers. Specifically, for each question q, we obtain multiple answers from the model. Following the practice of Cheng et al. (2024) and to balance statistical reliability with computational efficiency, we set the number of answers to 10. To automatically process the free-format answers generated by the model, inspired by Manakul et al. (2023), we adopt techniques from the Extractive Question Answering task (Chen et al., 2019). Specifically, we utilize a DeBERTa-v3-large (He et al., 2021) model fine-tuned on SQuAD2.0 (Rajpurkar et al., 2018) to extract core answers from free-format responses. This process standardizes diverse answer formats and improves response comparability.

After extracting core answers, we measure answer consistency by calculating the frequency of repeated responses. We define the frequency of the most repeated answer a, denoted as $\text{freq}(a)$, as a measure of the model's confidence in its response. The belief state is determined based on the following formula:

$$s = \begin{cases} s_{\text{con}} & \text{if } \text{freq}(a) \geq \delta_{con} \\ s_{\text{unc}} & \text{if } \text{freq}(a) < \delta_{unc} \end{cases}$$

where s is the model's belief state for question q. To more precisely distinguish belief states, we introduce two thresholds δ_{con} and δ_{unc} ($\delta_{\text{con}} > \delta_{\text{unc}}$): if the model generates highly consistent answers to a question, it indicates that the model has high confidence and stability in its own answer, corresponding to the confident state (s_{con}). Conversely, if the answers are dispersed and lack consistency, it suggests that the model has a high degree of uncertainty about its own answer, corresponding to the uncertain state (s_{unc}).

Algorithm 1 BAFH

Require: Question q, LLM Model E, Belief State Classifier T
Ensure: Hallucination type v: overconfident or unaware

```
/* Step 1: Answer Generation and hidden states Retrieval */
a ← E(q) // Generate answer a for question
H ← HiddenState(E, q, a) // Get hidden states H
/* Step 2: Hallucination Detection and Belief State Classification*/
r ← HallucinationDetection(q, a) // Detect hallucination by comparing a with external knowledge
s ← T(H) // Classify belief state for a
/* Step 3: Factuality Hallucination Classification */
if r = "Hallucination" then
    if s = s_con then
        v ← "Overconfident Hallucination"
    else if s = s_unc then
        v ← "Unaware Hallucination"
    end if
else
    return "No Hallucination"
end if
return v
```

Training Set Construction To obtain the model's hidden states during the generation process, we concatenate the question with the model's answer and extract the hidden states of the i-th token in the l-th layer, represented as $h^{l,i} \in \mathbb{R}^d$, where d is the dimension of the hidden states. These hidden states serve as input to the classifier, with the corresponding belief state s_i assigned as the label. This forms a training dataset of N samples $\left\{h_j^{l,i}, s_j\right\}_{j=1}^{N}$.

Classifier Training The belief state classifier employs a feedforward neural network with hidden layer sizes of 256, 128, and 64, all utilizing ReLU activations. The classifier takes hidden state vector $h^{l,i} \in \mathbb{R}^d$ as input and produces a binary label (confident/uncertain) through a sigmoid-activated output layer. The classifier does not rely on hyperparameters such as temperature or top-k, ensuring robustness and avoiding the resource-intensive need for multiple question-answering sessions required by self-consistency methods (Slobodkin et al., 2023; Su et al., 2024). Given that models, domains, and prompts influence consistency, we construct datasets specific to these factors and train dedicated classifiers accordingly.

To distinguish between overconfident hallucinations and unaware hallucinations, BAFH analyzes the model's belief state during answer generation. The detection framework combines the model's belief state with advanced hallucination detection methods to determine the factuality hallucination type. We present the algorithm flow for factuality hallucination type detection in Algorithm 1.

5 Experimental Setting

5.1 Dataset

To constructed dataset and evaluate the performance of BAFH, we considered three existing QA benchmarks as data sources:

TriviaQA (Joshi et al., 2017) is a reading comprehension dataset. Its question-answer pairs can be used for open-domain question-answer tasks.

NQOPEN (Kwiatkowski et al., 2019) is a question answering dataset consisting of real queries issued to the Google search engine.

ALCUNA (Yin et al., 2023a) is a benchmark to assess LLMs' abilities in new knowledge understanding.

Evaluation Metrics Our evaluation follows a similar approach to Cheng et al. (2024), with modifications to better suit our task. We employ the following four metrics:

OH (Overconfident Hallucination): The proportion of correctly detected overconfident hallucinations among all overconfident hallucinations.

UH (Unaware Hallucination): The proportion of correctly detected unaware hallucinations among all unaware hallucinations.

Truthful Rate: The overall proportion of hallucination types correctly detected.

In addition, we also use **AUC** (Area Under the Curve) as an evaluation metric. Note that AUC is not applicable to prompt-based methods, as they do not produce continuous confidence scores.

5.2 Baselines

Most prior work focuses on detecting the presence of hallucinations, while the identification of hallucination types remains underexplored. Therefore, we use the results of LLM's self-assessment of its own hallucination types as the baseline.

Directly providing hallucinated responses and asking the LLM is unreasonable because this task is too challenging for LLM. Therefore, we design a multiple-choice open-domain QA task to indirectly evaluate the model's ability to detect its own hallucination types. In this task, the model must choose from three options: its own *hallucinated response*, the *correct answer* to the question, and *I don't know*. Selecting *I don't know* or the *correct answer* indicates that the model recognizes its knowledge limitations, corresponding to Unaware Hallucination. Conversely, selecting its own hallucinated response suggests that the model remains confident in its answer, corresponding to Overconfident Hallucination. We use the model's performance on this task to measure its ability to perceive hallucination types and compare the performance with BAFH.

We adopt two prompting strategies:

Direct Instruction Prompt, where the model is directly instructed to select an answer.

Few-shot Prompt, which provides examples to illustrate the task requirements and then prompts the model to select the correct answer.

In both methods, we use greedy decoding to ensure determinism in the generated outputs, allowing for a more accurate assessment of the model's perception of hallucination types. The details of the prompts can be found in Appendix D.

To further evaluate the performance of the belief state classifier, we compare our method against the following uncertainty estimation approaches: (1) **MIND** is an unsupervised framework that leverages LLMs' internal states for real-time hallucination detection. (2) **SAR** (Duan et al., 2024) is one of the latest uncertainty estimation methods based on probability sampling and attention allocation.

5.3 Implementation Details

Dataset Construction To comprehensively evaluate the performance and generalizability of our factuality hallucination type detection method, we generate data using multiple open-source LLMs (including Gemma, Llama, and Mistral series) across various tasks. Following the procedure in Section 4.2, we utilize TriviaQA, NQOPEN, and ALCUNA as data sources to build model-specific datasets.

The training set contains 3,000 samples, evenly distributed between confident and uncertain states, which are used to train the belief state classifier.

Models	Methods	ALCUNA	NQOPEN				TriviaQA			
		Truthful	AUC	Truthful	UH	OH	AUC	Truthful	UH	OH
Gemma-2-27b-it	Direct Instruction	0.385	-	0.274	0.312	0.236	-	0.304	0.336	0.272
	Few-shot	0.47	-	0.294	0.352	0.236	-	0.338	0.398	0.278
	BAFH	**0.999**	**0.9063**	**0.821**	**0.854**	**0.788**	**0.8623**	**0.769**	**0.89**	**0.648**
Gemma-2-9b-it	Direct Instruction	0.643	-	0.31	0.36	0.26	-	0.33	0.38	0.28
	Few-shot	0.661	-	0.314	0.378	0.25	-	0.321	0.386	0.256
	BAFH	**0.992**	**0.8907**	**0.799**	**0.784**	**0.814**	**0.8406**	**0.751**	**0.836**	**0.666**
Gemma-2-2b-it	Direct Instruction	0.617	-	0.259	0.33	0.188	-	0.18	0.226	0.134
	Few-shot	0.638	-	0.306	0.37	0.242	-	0.217	0.31	0.124
	BAFH	**0.989**	**0.8601**	**0.766**	**0.75**	**0.782**	**0.8111**	**0.719**	**0.836**	**0.602**
Llama-3.1-70B-Instruct	Direct Instruction	0.55	-	0.327	0.436	0.218	-	0.34	0.476	0.204
	Few-shot	0.518	-	0.313	0.43	0.196	-	0.328	0.468	0.188
	BAFH	**0.877**	**0.7924**	**0.741**	**0.708**	**0.774**	**0.7509**	**0.675**	**0.688**	**0.662**
Llama-3.1-8B-Instruct	Direct Instruction	0.664	-	0.4	0.49	0.31	-	0.364	0.474	0.254
	Few-shot	0.84	-	0.526	0.656	0.396	-	0.481	0.64	0.322
	BAFH	**0.993**	**0.8605**	**0.771**	**0.824**	**0.718**	**0.7982**	**0.705**	**0.876**	**0.534**
Llama-3-70B-Instruct	Direct Instruction	0.406	-	0.371	0.42	0.322	-	0.377	0.448	0.306
	Few-shot	0.431	-	0.407	0.482	0.332	-	0.453	0.536	0.37
	BAFH	**0.894**	**0.7894**	**0.706**	**0.636**	**0.776**	**0.7894**	**0.709**	**0.71**	**0.708**
Llama-3.1-8B-Instruct	Direct Instruction	0.56	-	0.415	0.476	0.354	-	0.393	0.502	0.284
	Few-shot	0.618	-	0.451	0.554	0.348	-	0.41	0.524	0.296
	BAFH	**0.973**	**0.8117**	**0.722**	**0.678**	**0.766**	**0.7521**	**0.687**	**0.758**	**0.616**
Mistral-7B-Instruct-v0.3	Direct Instruction	0.553	-	0.397	0.506	0.288	-	0.363	0.45	0.276
	Few-shot	0.5	-	0.453	0.52	0.386	-	0.397	0.476	0.318
	BAFH	**0.907**	**0.8232**	**0.759**	**0.72**	**0.798**	**0.747**	**0.683**	**0.694**	**0.672**

Table 1: Performance comparison of different models and methods across multiple datasets and metrics

The training set focuses solely on the model's belief state regarding its answers.

The test set consists of 1,000 hallucination samples, evenly split into overconfident and unaware hallucinations, which is used to evaluate the accuracy of factuality hallucination type detection.

Notably, our datasets constructed from TriviaQA and NQOPEN include both training and test sets, while the dataset constructed from ALCUNA only includes a test set, for evaluating the performance of Unaware Hallucination detection.

Hidden States Selection In our main experiments, we use the model's last layer hidden states of the last token as features. This choice is based on findings from previous research (Azaria and Mitchell, 2023; Chuang et al., 2023), which suggest that the final layers tend to encode more abstract and high-level semantic information. Given that hidden states of different tokens in various layers may encode varying levels of semantic information, we analyze multiple token-layer combinations and compare their effects in the ablation study.

Threshold Selection In the main experiments, to ensure distinction between belief states and cover most of the data, we set δ_{con}=10 and δ_{unc}=5. In Section 6.3 we present a comparative analysis of different threshold settings.

6 Results

We conduct experiments to evaluate our proposed factuality hallucination type detection method. Specifically, this section aims to answer the following research questions (RQs):

RQ1: Does BAFH achieve good performance?

RQ2: How do the two types of hallucinations differ from each other and from correct answers?

RQ3: Can hidden state of LLMs be used to distinguish different types of hallucinations?

6.1 Overall Results of BAFH and Baselines

In this section, we conduct a comprehensive evaluation of the BAFH framework against baselines to address research question **RQ1**. Table 1 presents a comparison of BAFH with constructed baselines across eight LLMs and three QA datasets. Our findings are as follows:

(1) BAFH outperforms baselines across all models and datasets, demonstrating strong generalization, as further evidenced in Appendix C. This suggests LLMs exhibit distinct belief states when generating factual errors and leveraging LLM hidden

states allows us to infer the model's belief state and, consequently, the hallucination type.

(2) In most cases, prompt-based methods yield UH values below 50% across all datasets. This indicates that models tend to provide answers rather than acknowledge their knowledge limitations when responding to questions, which aligns with findings from previous studies (Yin et al., 2023b). Interestingly, the UH metric for prompt-based methods generally outperforms the OH metric across most models and datasets, suggesting that models more readily admit to being unaware but struggle to identify their own overconfident hallucinations. The Few-shot approach outperforms the Direct Instruction method, demonstrating that guiding the model with examples helps it recognize its own biases and limitations.

(3) Both BAFH and prompt-based methods perform better on the ALCUNA dataset compared to others, revealing differences in model belief states between new and existing knowledge.

(4) With classifier parameters fixed, the performance of the classifier varies with model size. In the Gemma series, the *Truthful* of classifier positively correlates with model size, possibly due to richer feature representations in the hidden states of larger models. In contrast, for the Llama series, *Truthful* decreases as model size increases, which may be because the classifier struggles to fully exploit the increasingly complex internal features beyond a certain scale.

Methods	Llama2-7B	Llama2-13B
BAFH	**0.758**	**0.794**
MIND	0.627	0.568
SAR	0.702	0.644

Table 2: The experimental results of BAFH and other baselines on our self-construct dataset based on TrivialQA

Performance of the Belief State Classifier As a key component of BAFH, the belief state classifier significantly impacts the framework's effectiveness. In this section, we compare it with state-of-the-art methods, MIND and SAR, on the dataset based on TriviaQA. MIND is as a strong representative of linear probing approaches and SAR is one of the most effective probability-based methods.

As shown in Table 2, BAFH outperforms both baselines in belief state classification. This may be because our dataset relies on self-consistency rather than correctness, which better aligns with hallucination classification by capturing the internal belief patterns of LLMs. Furthermore, leveraging hidden layer activations enables the classifier to capture more nuanced semantic representations. We also assess computational efficiency (Appendix B.2), showing that BAFH maintains competitive efficiency while achieving superior performance.

Models	Confident State	Uncertain State
Gemma-2-2b-it	0.8561	0.4973
Llama3-8B-Instruct	0.7892	0.4766
Llama3.1-8B-Instruct	0.8056	0.5105
Mistral-7B-Instruct	0.7279	0.4719

Table 3: Model's Hallucination Selection Rates in Multiple-Choice Questions for Overconfident and Unaware Hallucinations

6.2 The Difference Between the Two Hallucinations

We construct a multiple-choice task to address **RQ2**. The results are shown in Table 3. Specifically, we first extract the hallucinated answers generated by the model using the method described in Section 4.2. These answers could either be Overconfident Hallucinations or Unaware Hallucinations. We then form multiple-choice questions by presenting the model's hallucinated answer and the ground-truth answer as the two answer choices, with the original question serving as the prompt.

Since automatic extraction of answers may introduce errors, we conducted manual screening to ensure data quality, as detailed in Appendix B.1. Finally, we separately compute the hallucination selection rates for the two types of questions:

Confident State Group: The proportion of times the model selected its own hallucinated answer in all multiple-choice questions containing an Overconfident Hallucination.

Uncertain State Group: The proportion of times the model selected its own hallucinated answer in all multiple-choice questions containing an Unaware Hallucination.

The results show that LLMs tend to prefer their own answers when confident, while their choices appear random when uncertain. This may indicate that factuality hallucinations stem from different causes, such as inherent biases or a lack of relevant knowledge. These findings highlight the role of belief states in differentiating hallucination types.

Internal Space Differentiation To address **RQ2** and **RQ3**, we perform a PCA projection of the

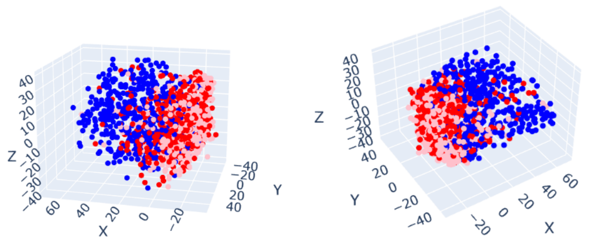


Figure 5: 3D PCA projection of the last hidden layer's embedding of LLaMA-3-8B-Instruct

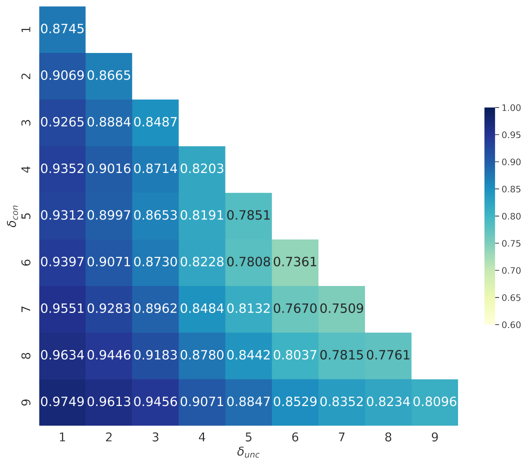


Figure 6: AUC of BAFH under different thresholds

embedding from the final hidden layer of the last generated token onto a 3-D plane. Figure 5 illustrate the results for LlaMA-3-8B-Instruct on the NQOPEN dataset. We observe that the boundary between Overconfident Hallucinations (pink dots) and Correct Answers (red dots) is not distinct. Furthermore, Unaware Hallucinations (blue dots) form a distinguishable, though not sharply defined, boundary with the other two categories. This suggests that the model's hidden states can be used to differentiate between different belief states, and overconfident hallucinations show a strong similarity to correct answers in their belief states.

6.3 Ablation Studies

Effect of δ_{con} and δ_{unc} threshold We investigate the impact of belief state thresholds δ_{con} and δ_{unc} on the model's AUC metric. To mitigate the influence of data distribution, we constructe balanced datasets for training and testing under various threshold combinations. The results are illustrated in Figure 6. As the gap between δ_{con} and δ_{unc} thresholds widens, the classifier's AUC improves significantly. This indicates that larger threshold differences better capture variations in belief states. Additionally, the consistency level of answers reflects the model's belief state, with higher consistency suggests greater model confidence in its responses.

Layers	Token Positions	
	Qend	Aend
20	0.7818	0.8901
24	0.7612	0.8867
28	0.7585	0.8871
32	0.7703	0.8848

Table 4: AUC scores across different token positions and layers

Token and Hidden Layer Selection To examine the impact of token position and hidden layer selection on framework performance, We conduct experiments using data generated by Llama3-8B-Instruct on the NQOPEN dataset. We focus on tokens at the question's end (Qend) and the sequence's end (Aend), as well as hidden layers near the output. As shown in Table 4, tokens at the same position perform similarly across different layers, whereas classification accuracy is significantly affected by token position. The sequence-end token (Aend) performs best, likely due to its hidden states retaining more belief state-related information.

7 Conclusion

We propose a belief-state-based factuality hallucination classification method and introduce BAFH, a hallucination type detection method. Experimental results show that BAFH achieves high accuracy across multiple datasets. Furthermore, different types of hallucinations are distinct in the distribution of hidden states, and LLMs exhibit distinct behavioral patterns when encountering different hallucination types. However, LLMs struggle to recognize the hallucination types of their own. In summary, our research reveals distinctions among factuality hallucination categories and highlights the significance of hallucination classification.

8 Limitations

This study focuses on the classification of factuality hallucinations, while more challenging types, such as faithfulness hallucinations and those involving complex reasoning, have not been explored in depth. Future work will incorporate a broader range of hallucination types and classification criteria to provide a more comprehensive understanding of the differences between them.

Meanwhile, this study primarily aims to identify hallucination types and analyze their differences, rather than directly investigating the causes of hallucinations or the key factors influencing their types. We believe that hallucination classification can contribute to understanding the mechanisms behind hallucination generation and lay the groundwork for future research on its causes and influencing factors. This direction will be further explored in our future work.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Amos Azaria and Tom Mitchell. 2023. The internal state of an LLM knows when it's lying. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 967–976, Singapore. Association for Computational Linguistics.

Anthony Chen, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. Evaluating question answering evaluation. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 119–124, Hong Kong, China. Association for Computational Linguistics.

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

I-Chun Chern, Steffi Chern, Shiqi Chen, Weizhe Yuan, Kehua Feng, Chunting Zhou, Junxian He, Graham Neubig, and Pengfei Liu. 2023. Factool: Factuality detection in generative ai – a tool augmented framework for multi-task and multi-domain scenarios. *Preprint*, arXiv:2307.13528.

Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James Glass, and Pengcheng He. 2023. Dola: Decoding by contrasting layers improves factuality in large language models. *arXiv preprint arXiv:2309.03883*.

Roi Cohen, May Hamri, Mor Geva, and Amir Globerson. 2023. LM vs LM: Detecting factual errors via cross examination. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12621–12640, Singapore. Association for Computational Linguistics.

Jinhao Duan, Hao Cheng, Shiqi Wang, Alex Zavalny, Chenan Wang, Renjing Xu, Bhavya Kailkhura, and Kaidi Xu. 2024. Shifting attention to relevance: Towards the predictive uncertainty quantification of free-form large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5050–5063, Bangkok, Thailand. Association for Computational Linguistics.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*.

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

Yichong Huang, Xiachong Feng, Xiaocheng Feng, and Bing Qin. 2023b. The factual inconsistency problem in abstractive text summarization: A survey. *Preprint*, arXiv:2104.14839.

Ziwei Ji, Delong Chen, Etsuko Ishii, Samuel Cahyawijaya, Yejin Bang, Bryan Wilie, and Pascale Fung. 2024. Llm internal states reveal hallucination risk faced with a query. *Preprint*, arXiv:2407.03282.

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

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. triviaqa: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. *arXiv e-prints*, arXiv:1705.03551.

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

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

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

Junyi Li, Xiaoxue Cheng, Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2023. 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, Singapore. Association for Computational Linguistics.

Wei Li, Wenhao Wu, Moye Chen, Jiachen Liu, Xinyan Xiao, and Hua Wu. 2022. Faithfulness in natural language generation: A systematic survey of analysis, evaluation and optimization methods. *Preprint*, arXiv:2203.05227.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022a. 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, Dublin, Ireland. Association for Computational Linguistics.

Stephanie C. Lin, Jacob Hilton, and Owain Evans. 2022b. Teaching models to express their uncertainty in words. *Trans. Mach. Learn. Res.*, 2022.

Junyu Luo, Cao Xiao, and Fenglong Ma. 2023. Zero-resource hallucination prevention for large language models. *Preprint*, arXiv:2309.02654.

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.

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.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

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.

Aviv Slobodkin, Omer Goldman, Avi Caciularu, Ido Dagan, and Shauli Ravfogel. 2023. The curious case of hallucinatory (un)answerability: Finding truths in the hidden states of over-confident large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3607–3625, Singapore. Association for Computational Linguistics.

Weihang Su, Changyue Wang, Qingyao Ai, Yiran Hu, Zhijing Wu, Yujia Zhou, and Yiqun Liu. 2024. Unsupervised real-time hallucination detection based on the internal states of large language models. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 14379–14391, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

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

Jia-Yu Yao, Kun-Peng Ning, Zhen-Hui Liu, Mu-Nan Ning, Yu-Yang Liu, and Li Yuan. 2024. Llm lies: Hallucinations are not bugs, but features as adversarial examples. *Preprint*, arXiv:2310.01469.

Xunjian Yin, Baizhou Huang, and Xiaojun Wan. 2023a. ALCUNA: Large language models meet new knowledge. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1397–1414, Singapore. Association for Computational Linguistics.

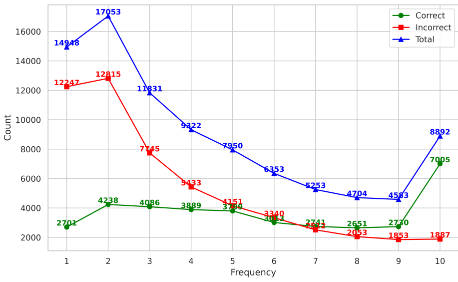
Zhangyue Yin, Qiushi Sun, Qipeng Guo, Jiawen Wu, Xipeng Qiu, and Xuan-Jing Huang. 2023b. Do large language models know what they don't know? In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8653–8665.

Tianhang Zhang, Lin Qiu, Qipeng Guo, Cheng Deng, Yue Zhang, Zheng Zhang, Chenghu Zhou, Xinbing Wang, and Luoyi Fu. 2023a. Enhancing uncertainty-based hallucination detection with stronger focus. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 915–932, Singapore. Association for Computational Linguistics.

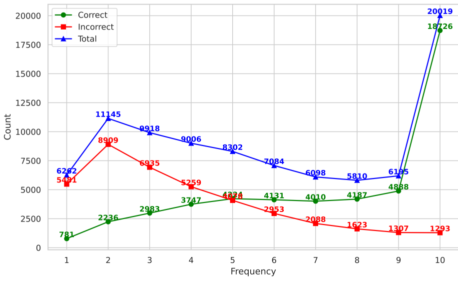
Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, and Shuming Shi. 2023b. Siren's song in the ai ocean: A survey on hallucination in large language models. *Preprint*, arXiv:2309.01219.

A Validation of the hypothesis

A.1 Statistical Analysis of Model Response Consistency

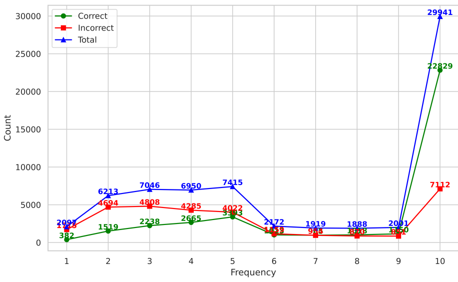


(a) NQOPEN

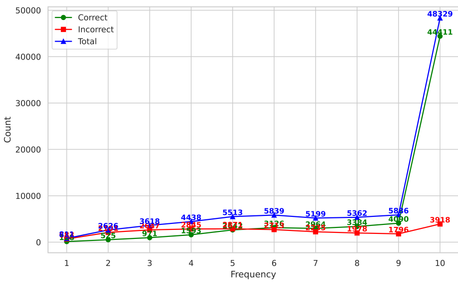


(b) TriviaQA

Figure 7: Gemma-2-2b-it

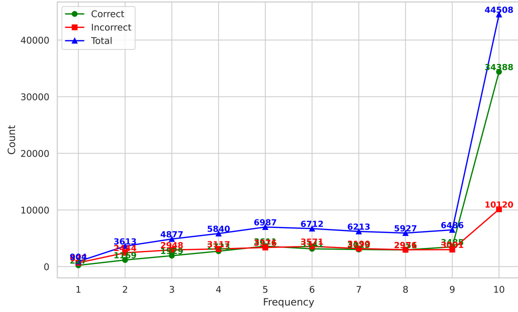


(a) NQOPEN

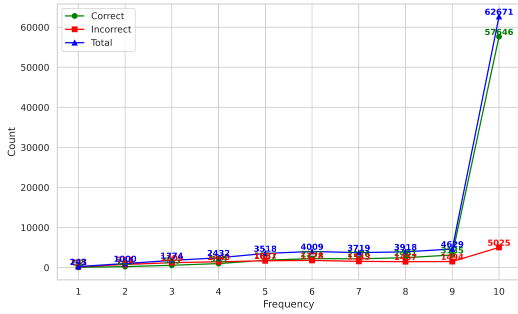


(b) TriviaQA

Figure 8: Gemma-2-9b-it

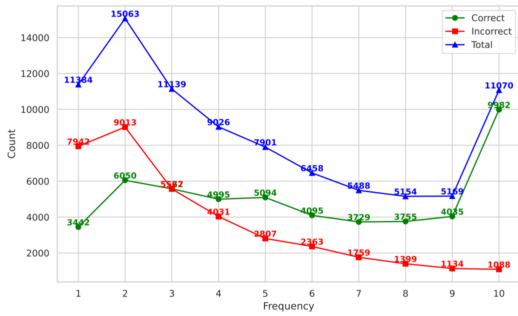


(a) NQOPEN

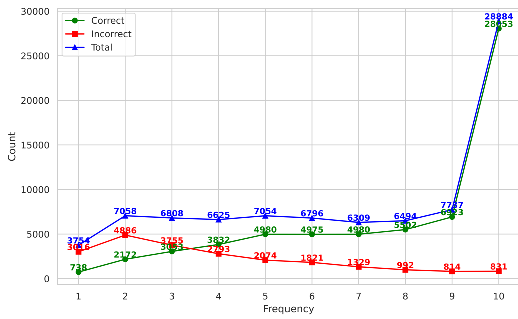


(b) TriviaQA

Figure 9: Gemma-2-27b-it

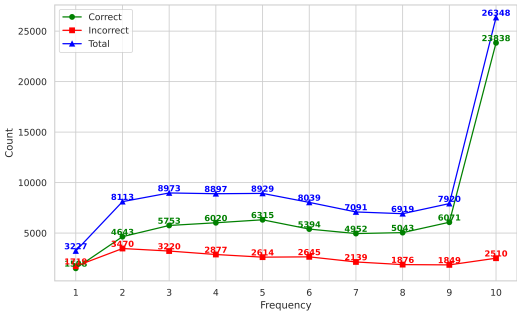


(a) NQOPEN

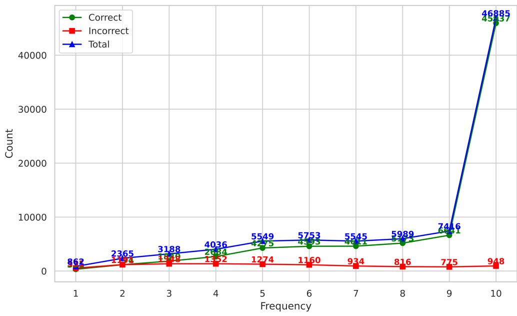


(b) TriviaQA

Figure 10: LLaMA-3.1-8B-Instruct



(a) NQOPEN



(b) TriviaQA

Figure 11: LLaMA-3.1-70B-Instruct

In this section, we conduct experiments on LLaMA-3.1 and Gemma-2 to analyze the repetition rate of model responses. As shown in Figre 7 to 11. The results align with our hypothesis: the distribution of response repetition rates is uneven, with higher and lower repetition rates being more prevalent, while intermediate repetition rates are relatively less frequent. Moreover, similar patterns are observed across other models. The bimodal phenomenon is more pronounced in smaller models but less apparent in larger ones. This may be because the dataset used is relatively simple for the larger models, leading to more high-confidence and high-accuracy predictions, while uncertain cases are relatively rare.

A.2 Internal Space Differentiation

In this section, we visualize the internal states of the model's hallucinations. As shown in Figure 12, blue points represent hallucinations with high repetition counts (9-10), red points represent those with low repetition counts (1-4), and green points represent hallucinations with intermediate repetition counts. The results indicate that the internal states of hallucinations with high and low repetition counts exhibit separation, whereas hallucinations with intermediate repetition counts do not form a distinct category, suggesting that their belief states are difficult to classify into a separate group.

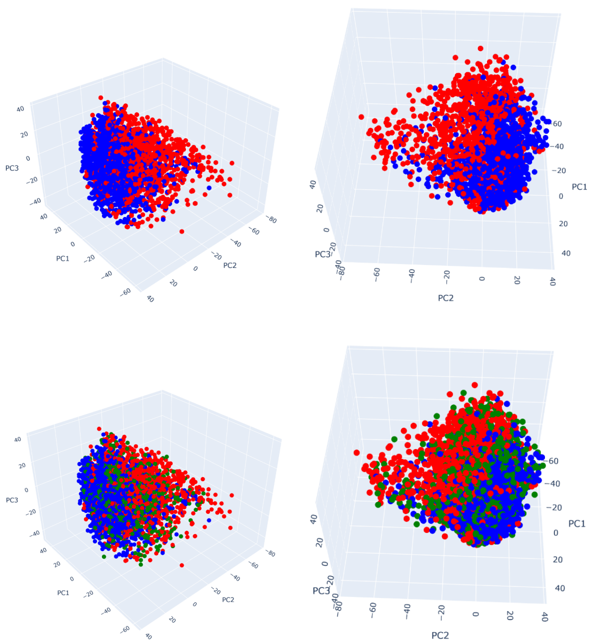


Figure 12: 3D PCA projection of the last hidden layer's embedding of LLaMA-3.1-8B-Instruct

A.3 A more fine-grained classification method.

Number of Classes	2	3	4
Llama3-8B-Instruct	0.7703	0.3395	-
Gemma_2_9b_it	0.7325	0.3226	-

Table 5: Performance comparison of Llama3-8B-Instruct and Gemma_2_9b_it with different numbers of classes.

In this section, we attempt to train the linear classifier using hidden states to categorize belief states at a finer granularity and evaluate its F1 scores under different numbers of categories. As shown in Figure 5, the binary classification setting achieves the best performance, while in the three-class setting, the classifier's performance is close to random. In the four-class setting, the classifier struggles to converge effectively, indicating a high degree of uncertainty in the task.

This phenomenon may be due to the fact that finer-grained classification of belief states is more susceptible to various potential noise factors, which in turn affect the classifier's performance. Therefore, dividing belief states into two categories is a reasonable simplification.

B Implementation details

B.1 Construct multiple-choice questions.

We first use ChatGPT to perform an initial filtering of hallucination types that meet the definition. Then, we invite a human annotator to further refine the selection in order to construct high-quality multiple-choice questions containing both types of hallucinated responses from the model. Specifically, we obtain an initial hallucination type dataset following the process outlined in Section 4.2, after which ChatGPT conducts a preliminary screening. A human annotator then reviews the dataset, ensuring the correctness of the hallucination type labels from the following three aspects: the answer is correctly extracted from the model's response, it meets the hallucination definition, and the answer's repetition rate is calculated correctly. For each model in Table 3, we ultimately retain 1000 multiple-choice questions that meet the requirements, with 500 questions for each type of hallucination.

Method	**Train Time (s)**	**Inference Time (s)**
LLM's Response	–	1.52
BAFH	17.90	0.05
MIND	18.47	0.05
SAR	–	<0.01

Table 6: Comparison of training and inference times for BAFH and other baselines using Llama-7B hidden activations.

B.2 Computational Cost of the Belief State Classifier

Table 6 shows a comparison of the training and inference times for the BAFH method versus other baselines using hidden layer activations of Llama-7B. The experiment was conducted on an NVIDIA V100 GPU. The training time of our method is comparable to the hidden layer activation-based method MIND[3], and it is significantly faster than the response time of LLMs.

C Generalization Experiments

We evaluate the generalization capability of our approach. We train the classifier using a balanced training set based on NQOPEN and test it on a test set derived from TriviaQA. As shown in Table 7, the classifier trained on NQOPEN data maintains good performance on the out-of-domain test set, demonstrating the strong generalization ability of our detection framework. This effectiveness may be attributed to two factors: First, the belief state classifier, trained on hidden states, is less influenced by textual features. Second, it appears to successfully capture the model's underlying belief state. These results suggest that our approach can generalize well across different domains and datasets.

Model ID	NQOPEN-TriviaQA			
	AUC	Truthful	UH	OH
Llama3.1_70B_Instruct	0.7606	0.658	0.868	0.448
Llama3.1_8B_Instruct	0.8532	0.763	0.818	0.708
Llama3_70B_Instruct	0.7432	0.671	0.730	0.612
Llama3_8B_Instruct	0.8111	0.734	0.812	0.656
Mistral_7B_Instruct	0.7996	0.732	0.758	0.706
Gemma_2_27b_it	0.9027	0.797	0.940	0.654
Gemma_2_2b_it	0.8108	0.728	0.748	0.708
Gemma_2_9b_it	0.8476	0.758	0.862	0.654

Table 7: Performance comparison of different models on NQOPEN-TriviaQA dataset

D Prompt

I want you to act as a response judge. If you don't know the answer, or if you know that neither of the provided options is correct, you should answer "I don't know." If you know the answer to the question and the correct answer is among the provided options, you should choose the best and correct option. You must either select an option or answer "I don't know."

Here are some examples to illustrate how to make a choice:

Example 1:
Question: What is the capital of France?
Options: 1)Paris 2)Berlin
Your judgment: 1) Paris
Example 2:
…

Now, please follow these guidelines to answer the following question and options. You must either select one of the options or answer "I don't know."
Question: {question}
Options: {model output} {ground truth} (The order is random.)
Your judgment:

Figure 13: Few-shot prompt used for dataset construction

I want you to act as a response judge. If you don't know the answer, or if you know that neither of the provided options is correct, you should answer "I don't know." If you know the answer to the question and the correct answer is among the provided options, you should choose the best and correct option. You must either select an option or answer "I don't know."

Now, please follow these guidelines to answer the following question and options. You must either select one of the options or answer "I don't know."
Question: {question}
Options: {model output} {ground truth} (The order is random.)
Your judgment:

Figure 14: Direct Instruction used for dataset construction

Here are some examples to illustrate how to make a choice:

You should try your best to select the best and correct answer. If the two answers are the same, you can randomly choose one. If both answers are incorrect, choose the better one. You MUST select an answer from the provided two answers.

Here are some examples:

Example 1:
Question: Who came up with the vaccine for smallpox?
Answer1: Edward Jenner
Answer2: Jennifer Aniston
Your Choice: The best answer is Answer1, Edward Jenner.
Example 2:
Question: Where is the headquarters of geological survey of india located?
Answer1: Delhi
Answer2: Kolkata
Your Choice: The best answer is Answer2, Kolkata.

Actual Task

Question: {Question}
Answer1: {ground truth}
Answer2: {Overconfident Hallucination / Unaware Hallucination}
Your Choice:

Figure 15: Prompts used for multiple-choice task to address **RQ2**