Principled Detection of Hallucinations in Large Language Models via Multiple Testing

Jiawei Li[*1], **Akshayaa Magesh**[*2], **Venugopal V. Veeravalli**[1]
[1]University of Illinois Urbana-Champaign, [2]Meta
{jiaweil9,vvv}@illinois.edu, akshayaamagesh@gmail.com

Abstract

While Large Language Models (LLMs) have emerged as powerful foundational models to solve a variety of tasks, they have also been shown to be prone to *hallucinations*, i.e., generating responses that sound confident but are actually incorrect or even nonsensical. Existing hallucination detectors propose a wide range of empirical scoring rules, but their performance varies across models and datasets, and it is hard to determine which ones to rely on in practice or to treat as a reliable detector. In this work, we formulate the problem of detecting hallucinations as a hypothesis testing problem and draw parallels with the problem of out-of-distribution detection in machine learning models. We then propose a multiple-testing-inspired method that systematically aggregates multiple evaluation scores via conformal p-values, enabling calibrated detection with controlled false alarm rate. Extensive experiments across diverse models and datasets validate the robustness of our approach against state-of-the-art methods.

1 Introduction

Large language models (LLMs) (Caruccio et al., 2024; Team et al., 2024; Achiam et al., 2023) have emerged as powerful tools for a variety of tasks, most commonly generating answers to user-specified prompts, including text generation, summarization, and question answering (Keskar et al., 2019; Raffel et al., 2020; Zhang et al., 2024; Singhal et al., 2025). Although LLMs have shown strong capabilities in these applications, they have also been shown to be prone to *hallucinations*, i.e., generating responses that sound confident, but are actually incorrect, or even nonsensical (Ji et al., 2023; Yao et al., 2023; Zhang et al., 2023). Given the increasing reliance on LLMs in real-world scenarios, it is imperative to develop methods to detect whether an LLM is generating hallucinations for a given prompt.

The term 'hallucination' is general, and it captures different kinds of incorrect generations induced by different causes. For example, hallucinations can be classified as factual hallucinations and faithful hallucinations based on the error in the words (Huang et al., 2025). The causes of hallucinations vary, such as LLMs not learning the knowledge to answer the questions (Onoe et al., 2022), LLMs being trained on biased data (Ladhak et al., 2023), or LLMs producing sycophancy resulting from reinforcement learning from human feedback (RLHF) training models (Sharma et al., 2023). Apart from these, confabulations, i.e., arbitrary and possibly wrong generations from the language model caused by sensitivity to hyperparameters such as random seed, are also an important factor that can lead to incorrect answers in LLMs (Farquhar et al., 2024).

Various approaches have been proposed on hallucination detection in LLMs. External knowledge retrieval approaches (Chern et al., 2023; Huo et al., 2023) contrast model outputs with external databases to flag factual inconsistencies. Methods leveraging natural language inference (NLI) frameworks (Zhou et al., 2020) assess the consistency between generated content and canonical answers or reference facts, providing another lens to evaluate hallucinations. More recently, the 'LLM-as-a-judge' paradigm has emerged, wherein fine-tuned LLMs are tasked with judging the veracity of their outputs. Some studies have explored direct confidence scoring of model generations (Luo et al., 2023), while others have proposed frameworks that design prompts with diagnostic questions to probe hallucinations (Manakul et al., 2023; Muhammed et al., 2025; Yang et al., 2025b). Complementing these approaches, uncertainty estimation methods have been introduced to quantify the inherent ambiguity in model predictions a factor often observed

[*]Equal contribution.

Findings of the Association for Computational Linguistics: ACL 2026, pages 34132–34145
July 2-7, 2026

in hallucinated outputs (Varshney et al., 2023; Manakul et al., 2023; Rateike et al., 2023).

It remains unclear whether any one of these evaluation methods is sufficient to detect all classes of hallucinations. Hallucination patterns can manifest in different ways, making it unlikely that a single metric can detect hallucinations across diverse datasets and models. Additionally, given the rapid pace with which new LLMs are being developed, it is not realistic to develop specific hallucination detection methods for each new LLM. Therefore, it is of interest to design a generally robust method for hallucination detection that can leverage the advantages of preexisting methods, without any additional assumptions on the specific datasets or the LLMs being considered.

In this work, we build a unified multiple testing framework for hallucination detection that systematically combines scores proposed in prior work. The framework is agnostic to how the underlying scores are obtained; in principle, it can incorporate any collection of baseline detectors. In our experiments, however, we focus on training-free, retrieval-free scores that require no external factual repositories and no auxiliary judge models. Concretely, we leverage state-of-the-art uncertainty and consistency metrics derived from the models output layer by resampling multiple generations under different random seeds. For example, Kuhn et al. (2023) propose semantic entropy, which leverages meaning-invariant clustering to quantify uncertainty, and Lin et al. (2023) propose confidence measures computed from pairwise similarity among sampled generations, including lexical overlap and spectral scores based on a similarity graph. Finally, our framework adds a lightweight calibration step that uses a small calibration set of non-hallucinated prompts to provide theoretical control of the false-alarm rate.

1.1 Our contributions

We develop a robust method for hallucination detection by framing it as a hypothesis testing problem. Motivated by recent advances in out-of-distribution (OOD) detection, we propose adapting the principled detection procedure developed by (Magesh et al., 2023) to the problem of hallucination detection in LLMs. Specifically, we introduce a method that systematically integrates multiple evaluation scores using conformal p-values. Our key contributions are summarized as follows.

1. A hypothesis-testing-based framework. We reconceptualize hallucination detection as a hypothesis testing problem, drawing parallels with OOD detection in machine learning (Section 2). This provides a statistically grounded framework for identifying hallucinations in LLM-generated content.

2. A multiple-testing-inspired detection pipeline. We provide a non-trivial conceptual bridge that adapts multiple hypothesis testing to the problem of LLM hallucination. Unlike existing ad-hoc heuristics, our framework systematically manages the dependencies among heterogeneous signals, ranging from lexical overlap to deep semantic and spectral properties. This represents a significant shift toward principled inference, providing the first framework capable of offering calibrated false-alarm control in a generative context (Section 3).

3. Empirical validation across diverse datasets and models. We conduct extensive experiments across various LLM architectures and datasets (Section 4), demonstrating that our method maintains robustness across different datasets and LLMs, and generally outperforms existing hallucination detection techniques. Specifically, our method exhibits consistently high AUROC and detection power across LLaMA-2, LLaMA-3, Mistral, and DeepSeek-v2, on HaluEval, CoQA and TriviaQA (Table 1), whereas state-of-the-art methods exhibit high variability among different models and datasets. Moreover, when we macro-average AUROC over the full evaluation suite (spanning all tested models and datasets, including additional models such as Qwen3 variants and additional benchmarks such as FactCHD and GSM8K), our method attains the highest average performance, outperforming both prior baselines and simple aggregation strategies, such as majority voting and averaging (Figure 2). These empirical results corroborate the effectiveness and robustness of our proposed approach.

2 Problem Modeling

In this section, we first review existing methods in the literature for detecting hallucinations. We then define the problem from the perspective of hypothesis testing and introduce the multiple testing framework to leverage baseline methods. The objective is to distinguish prompts that are less likely to generate hallucinations, labeled as *correct*, from

prompts likely to generate hallucinations, labeled as *incorrect*.

2.1 Existing hallucination detection methods

In this part, we introduce several threshold-based methods to detect hallucinations in gray-box settings (with access to output likelihoods) or black-box settings (with access to the generations only). All these methods develop metrics to measure the uncertainty or similarity of multiple generations given the same prompt, and declare a prompt likely to generate hallucinations based on an empirical threshold applied to the measured metric.

Semantic Entropy. (Farquhar et al., 2024) consider the entropy of model generations under semantic clustering, where generations with the same semantic meaning are grouped into the same cluster. The semantic entropy score is calculated as

$$\mathbf{SE}(x) = -\sum_{i=1}^{|C|} \mathbb{P}(C_i \mid x) \log \mathbb{P}(C_i \mid x),$$

where x denotes the prompt, C_i denotes the i-th semantic cluster, obtained via semantic equivalence clustering based on bidirectional entailment, and $\mathbb{P}(C_i \mid x)$ is the normalized probability mass assigned to cluster C_i conditioned on x.

Alpha Semantic Entropy (αSE). (Kaur et al., 2024) also focus on semantic similarity among different generations, but provide a different algorithm to calculate the clustering of semantic equivalence, inspired by the distance-dependent Chinese restaurant process.

Spectral Eigenvalue. Instead of considering the semantic similarity problem as a black-and-white problem, (Lin et al., 2023) consider the semantic similarity between different samples as continuous real numbers and translate them into weights in a graph. The eigenvalues of the symmetric normalized graph Laplacian are then calculated from the symmetric weighted adjacency matrix. Suppose the eigenvalues are $\lambda_1 < \cdots < \lambda_m$, then the spectral eigenvalue score is given by

$$\mathbf{EigV}(x) = \sum_{i=1}^{m} \max(0, 1 - \lambda_i).$$

Kernel Semantic Entropy. (Nikitin et al., 2024) kernelize semantic entropy using the pairwise semantic similarities among M sampled generations. Let s_{ij} be the symmetric NLI entailment score between generations i and j, define $K_{ij} = \exp\big(s_{ij}/\tau\big)$ with $\tau > 0$, and use weights $w_j \geq 0$ with $\sum_{j=1}^{M} w_j = 1$. Let $p_i = \sum_{j=1}^{M} w_j K_{ij}$. The kernel semantic entropy is

$$\mathbf{KSE}(x) = -\sum_{i=1}^{M} w_i \log p_i.$$

Larger KSE indicates greater semantic dispersion (less clustering) among generations.

Lexical Similarity (LS). The lexical similarity score uses the sum of Rouge-L similarity scores among different samples in the generation, which is proposed in (Fomicheva et al., 2020).

2.2 Multiple hypothesis testing framework

Consider the question answering scenario where, given an input (prompt) X, the goal is to predict the output (answer) Y, which follows a ground truth distribution given X, $Y \sim P(Y|X)$. [1] In practice, Large Language Models (LLMs) are utilized to approximate this distribution by generating text conditioned on the input X. Let the model be denoted as $f(\mathbf{W}, \cdot)$, where $\mathbf{W}$ represents the parameters of the LLM. Given an LLM and test data $\mathcal{T} = \{(x_i, y_i)\}_{i=1}^{n}$, where x_i is the prompt and y_i is the reference answer, the objective is to detect whether the LLM may generate incorrect responses (hallucinations) for the given prompt x. The hallucination detection problem can be posed as a hypothesis testing problem as follows,

$$\begin{aligned} H_0 &: X \text{ is likely not to generate a hallucination} \\ &\quad \text{at the output of model } f(\mathbf{W}, X), \\ H_1 &: X \text{ is likely to generate a hallucination} \\ &\quad \text{at the output of model } f(\mathbf{W}, X), \end{aligned}$$

where H_0 is the null hypothesis, and H_1 is the alternative hypothesis.

As in general hypothesis testing settings, there are two metrics for performance evaluation, false alarm rate $\mathrm{P_F}$ and detection power $\mathrm{P_D}$. The false alarm rate indicates the probability of misclassifying correct generations from a given prompt as a hallucination, while the detection power is the

[1] Throughout the paper, uppercase characters (e.g., X and Y) denote random variables, while their realizations are denoted by lowercase characters (e.g., x and y).

probability of correctly detecting hallucinations. [2]

$$\mathrm{P_F} = \mathbb{P}_{H_0}(\text{declare hallucination}),$$
$$\mathrm{P_D} = \mathbb{P}_{H_1}(\text{declare hallucination}).$$

As described in the previous subsection, there are works that develop multiple scores for detecting hallucinations. Drawing ideas from OOD detection, we propose a multiple testing framework for hallucination detection, where we combine these scores from prior work. Specifically, given the model $\mathbf{W}$ and the input X, a list of scores T^i can be obtained by the score functions,

$$T^1 = s^1(X), T^2 = s^2(X), \cdots, T^K = s^K(X),$$

where $s^i(\cdot)$ depends on the model parameters $\mathbf{W}$. Since X is a random variable, T^i is also a random variable, and we denote the realization of T^i as t^i. The multiple testing framework is as follows:

$$\begin{aligned} H_{0,1} &: T^1 \sim P^1, & H_{1,1} &: T^1 \not\sim P^1; \\ &\vdots & &\vdots \\ H_{0,K} &: T^K \sim P^K, & H_{1,K} &: T^K \not\sim P^K, \end{aligned}$$

where $P^1, P^2, \cdots, P^K$ are the induced distributions of the scores for corresponding prompts for which the LLM does not generate hallucinations. We declare a generation from a given prompt to be a hallucination if any of $H_{0,i}$ is rejected, that is, if the *global null* is rejected.

Furthermore, since the distributions $P^1, P^2, \cdots, P^K$ are unknown, an additional calibration dataset $\mathcal{C} = \{x_i^c\}_{i=1}^{n_{\text{cal}}}$ is utilized to provide information about the prompts that lead to correct generations.

3 Proposed Methodology

In this section, we present our proposed algorithm for detecting hallucinations in the generations of LLMs. First, we compare the problem of detecting hallucinations in LLMs with the problem of out-of-distribution (OOD) detection in machine learning models, highlighting parallels and differences in Subsection 3.1. Then, in Subsection 3.2, we present our method for detecting hallucinations in LLM generations, adapted from the OOD detection procedure in (Magesh et al., 2023).

[2]Here $\mathbb{P}_{H_i}(\text{declare hallucination})$ means that, given the model $\mathbf{W}$ and prompt X, under the hypothesis H_i, the probability that the method declares that the prompt X will generate hallucinations.

3.1 Out-of-distribution detection and hallucination detection

A multiple hypothesis testing approach for out-of-distribution detection was first proposed in (Magesh et al., 2023). Here, we adapt this approach for hallucination detection. The two problems share several key similarities. They both aim to detect untrustworthy predictions or generations to improve robustness for the safe deployment of these models. Neither problem setting imposes assumptions on the distributions under the null and alternative hypotheses, because the input distribution is unknown when utilizing machine learning models during the test period. Additionally, both operate in a zero-resource setting, i.e., no additional data from the test distribution or extra training is used for detection.

Despite these similarities, there are key differences between hallucination detection and OOD detection as follows.

Indicator scores. Unlike standard OOD detection, which typically operates in a white-box setting by utilizing internal activations from multiple intermediate layers to capture fine-grained distribution shifts (Lee et al., 2018; De la Jara et al., 2025), our framework is designed for grey-box and black-box regimes. While OOD methods often derive scores from distance metrics within a unified latent space, LLM hallucination detection must bridge heterogeneous signals, ranging from lexical overlap to deep semantic entropy and spectral properties. Consequently, our scores are derived solely from final-layer output likelihoods or sampled generations, which are influenced by hidden parameters but do not directly expose them. This methodological choice is driven by the extreme scale of LLMs, where the vast number of parameters makes computing scores based on hidden-layer outputs costly or even infeasible. Such a limitation is especially critical for closed-source models, where the proprietary nature does not expose internal parameters, making our reliance on output-level signals essential for real-world deployment.

Data drawn from null hypothesis and alternative hypothesis. In OOD detection, the model is typically trained on one dataset and tested on another dataset, since small models can be efficiently trained and evaluated on different datasets, enabling clear hypothesis separation. The data from the training dataset serve as data from the null

hypothesis, and the data from another dataset serve as data from the alternative hypothesis. However, LLMs are pretrained on massive corpora at a high computational cost, and hallucination detection is typically performed on a fixed pretrained model without further training. Moreover, since LLMs generate outputs probabilistically from logits, different outputs can be produced for the same input, making it difficult to explicitly label data as coming from the null or alternative hypothesis. In this work, we adopt the Rouge-L score between generations and reference answers to label the generations as non-hallucinated or hallucinated. To accommodate potential rephrasing, we regard the input as not generating hallucinations if only a small fraction of sampled generations are classified as hallucinations among all the generations.

3.2 Proposed hallucination detection method

The multiple hypothesis testing algorithm is based on the general version of the Benjamini-Hochberg (BH) procedure that allows the scores to be dependent (Benjamini and Yekutieli, 2001). If the distribution of scores under the null hypothesis is known, the p-values of the observed score t^j_{test} can be computed as

$$q^j := \mathbb{P}_{H_0}(T^j \geq t^j_{\text{test}}),$$

with corresponding random version for a random test score T^j_{test} being denoted by Q^j.

However, in our hallucination detection problem, we do not have the distribution of scores under H_0, but we have access to the calibration set $\mathcal{C}$, which contains prompts that do not generate hallucinations. The dataset $\mathcal{C}$ can be used to compute empirical versions of the p-values, known as *conformal* p-values, of the scores. Denote the scores in the calibration set as $\{s^j_i = s^j(x^c_i) \ : \ x^c_i \in \mathcal{C}, j = 1, 2, \cdots, K\}$. Given the test scores t^j_{test} and the corresponding random variables T^j_{test}, the conformal p-values and their random versions, conditioned on the calibration dataset $\mathcal{C}$, are defined as

$$q^j_{\text{con}} := \frac{1 + |\{i : s^j_i \geq t^j_{\text{test}}\}|}{1 + |\mathcal{C}|},$$

$$Q^j_{\text{con}} := \frac{1 + |\{i : s^j_i \geq T^j_{\text{test}}\}|}{1 + |\mathcal{C}|}.$$

Algorithm 1 describes the method inspired by the BH procedure. The hyperparameter ϵ is related to the concentration property of the CDF of the Beta distribution, as detailed in Remark 3.1. A higher score indicates a higher likelihood of being classified as a hallucination. The conformal p-values for the test data are computed using the scores from the calibration dataset $\mathcal{C}$ and the scores from the test generations. These conformal p-values are then compared against ranked thresholds for hallucination detection. Theoretical guarantees for the false alarm rate can be obtained through the following theorem.

Algorithm 1: Multiple Hypothesis Testing for Hallucination Detection

Input: Test prompt with generations x_{test}; desired false alarm rate α; hyperparameter ϵ; calibration dataset $\mathcal{C}$ and its scores $\{s^j_i = s^j(x^c_i) : x^c_i \in \mathcal{C},\ j = 1, \ldots, K\}$.

Output: Decision on whether the prompt with generations is hallucinated.

for $j \leftarrow 1$ **to** K **do**
  $t^j_{\text{test}} \leftarrow s^j(x_{\text{test}})$;
  $q^j_{\text{con}} \leftarrow \frac{1+\left|\left\{i:\ t^j_{\text{test}} \leq s^j_i\right\}\right|}{1+|\mathcal{C}|}$;
end
$\widehat{q}^1_{\text{con}}, \ldots, \widehat{q}^K_{\text{con}} \leftarrow$ sorted values of $\{q^j_{\text{con}}\}_{j=1}^K$ in ascending order;
if $\exists\, j \in \{1, \ldots, K\}$ *such that* $\widehat{q}^j_{\text{con}} \leq \frac{\alpha}{(1+\epsilon)\sum_{i=1}^K \frac{1}{i}} \cdot \frac{j}{K}$ **then**
  return *Hallucination*;
else
  return *No Hallucination*;
end

Proposition 3.1 (Theorem 2 in (Magesh et al., 2023))**.** *Let $\alpha, \delta \in (0, 1)$. Denote the calibration set as $\mathcal{C}$. When the size of the calibration set $|\mathcal{C}|$ is sufficiently large, for a new input X and a learning model $f(\mathbf{W}, \cdot)$, with probability $1 - \delta$, the false alarm rate of Algorithm 1 is bounded by α, i.e.,*

$$\mathrm{P_F}(\mathcal{C}) = \mathbb{P}_{H_0}(\text{declare hallucination} \mid \mathcal{C}) \leq \alpha.$$

Remark 3.1. *Let $\epsilon > 0$, and denote by K the number of scores to be integrated and by α the desired false alarm rate. As stated in Lemma 1 in (Magesh et al., 2023), the size of calibration dataset $|\mathcal{C}|$ is sufficiently large if for the given $\delta > 0$,*

$$\min_{j=1,2,\cdots,K} I_{(1+\epsilon)\mu_j}(a_j, b_j) \geq 1 - \frac{\delta}{K^2},$$

where $a_j = \lfloor \frac{(|\mathcal{C}|+1)\cdot\alpha}{(1+\epsilon)\sum_{i=1}^K \frac{1}{i}} \cdot \frac{j}{K} \rfloor$, $b_j = |\mathcal{C}| + 1 - a_j$, $\mu_j = \frac{a_j}{a_j + b_j}$, and $I_x(a, b)$ denotes the CDF of Beta distribution $\mathrm{Beta}(a, b)$ evaluated at x.

Importantly, the required calibration size depends only on the parameters $(\alpha, \epsilon, \delta, K)$ and is

independent of the specific learning problem being considered or the pair of dataset and model being tested. When the calibration set is small, ϵ can be increased to accommodate the reduced sample size, at the cost of a more conservative decision rule that may reduce the detection power.

Remark 3.2. *Algorithm 1 requires neither additional model training nor extra data beyond what the underlying baseline scores already use. A calibration set can be constructed from prompts in the training data that are deemed non-hallucinated. As a result, our approach introduces minimal overhead and does not impose additional data or computational requirements beyond those of the chosen baselines. Furthermore, Algorithm 1 is broadly applicable: given a collection of baseline scoring rules for a detection (or more general decision) task, it can aggregate them into a single calibrated test with controlled false alarm rate.*

Remark 3.3. *Unlike prior work providing marginal guarantees over random calibration sets, this new method focuses on the conditional false alarm rate, which can be controlled at all times as long as the size of the calibration dataset is sufficiently large. The calibration sets diversity and coverage mainly affect detection power (e.g., redundancy can reduce power), while the false alarm guarantee remains valid.*

The final part is to construct the calibration dataset, which requires a method to label whether a given prompt with generations is hallucinated or not. Using the reference answer, each prompt-generation pair is labeled based on the Rouge-L similarity, following the criteria in Lin et al. (2023). As shown in Algorithm 2, once the Rouge-L similarity scores are high for most generations, the prompt is deemed not likely to generate hallucinations, since its generations are approximately correct. Finally, the calibration dataset is constructed by sampling from prompts that do not generate hallucinations.

Algorithm 2: Assign hallucination labels to prompts

Input: Dataset $\mathcal{D}$ with generations $\mathcal{D}_{p,j}$ for prompt p, reference answers Y with Y_p, similarity threshold τ, tolerance $\theta \in [0, 1]$.
Output: Prompt sets $\mathcal{D}_{\mathrm{NH}}$ (non-hallucinated) and $\mathcal{D}_{\mathrm{H}}$ (hallucinated).

$\mathcal{D}_{\mathrm{NH}}, \mathcal{D}_{\mathrm{H}} \leftarrow \emptyset, \emptyset$;
for $p \leftarrow 1$ **to** n_{prompts} **do**
  $h \leftarrow |\{j \in [n_{\mathrm{gen}}] : \text{Rouge-L}(\mathcal{D}_{p,j}, Y_p) \leq \tau\}|$;
  if $\frac{h}{n_{\mathrm{gen}}} \leq \theta$ **then**
    $\mathcal{D}_{\mathrm{NH}} \leftarrow \mathcal{D}_{\mathrm{NH}} \cup \{p\}$;
  else
    $\mathcal{D}_{\mathrm{H}} \leftarrow \mathcal{D}_{\mathrm{H}} \cup \{p\}$;
  end
end
return $(\mathcal{D}_{\mathrm{NH}}, \mathcal{D}_{\mathrm{H}})$;

4 Experimental Results

We demonstrate that our method shows robustness consistently across different datasets and language models. That is, regardless of the tested LLM or dataset, our method achieves consistently strong performance on hallucination detection. In contrast, other baseline scores often degrade in performance when faced with particular modeldataset combinations, or fail to outperform our method consistently. This highlights the reliability and generalizability of our approach, especially in real-world scenarios where user queries come from unknown and highly variable distributions.

Experimental setup. Based on the 20 sampled generations given the input, the Rouge-L score threshold, τ, is set to 0.3 to determine whether a generation is considered a hallucination. For a prompt, if at least 18 out of the 20 sampled generations ($\theta = 0.1$) are judged as non-hallucinations under Rouge-L evaluation, the prompt is considered not to generate hallucinations. We repeat the experiments 10 times using a randomized calibration dataset (a subset of prompts with non-hallucinated generations), and report the mean and standard deviation of our evaluation metrics. By default, the size of the calibration dataset is $1,000$. We test on models such as LLaMA-2-13B (Touvron et al., 2023), Mistral-7B (Jiang et al., 2023), Llama-3.1-8B (Grattafiori et al., 2024), DeepSeek-v2-Lite (DeepSeek-AI, 2024), Qwen3-4B (Yang et al., 2025a), Llama-3.2-3B-Instruct (Grattafiori et al., 2024), and Qwen2.5-Math-1.5B-Instruct (Yang et al., 2024b). Following (Kuhn et al., 2023), we evaluate our procedure on the HaluEval (Li et al., 2023), TriviaQA (Joshi et al., 2017), CoQA (Reddy et al., 2019), and GSM8K (Cobbe et al., 2021).

Baselines. We adopt five scoring functions to quantify the degree of hallucination induced by a prompt (Section 2.1). In addition, since the clustered variant of semantic entropy based on the frequencies of cluster assignments (Farquhar et al., 2024) can sometimes outperform the original formulation, we also report results for this clustered version.

Performance evaluation. Following prior work,

we report Area Under the Receiver Operating Characteristic curve (AUROC) in the main text; AUROC summarizes the tradeoff between detection power and false-alarm rate over all decision thresholds. In the appendix, we additionally report the detection power at a fixed 10% false alarm rate. Mathematically, given the calibration dataset $\mathcal{C}$, the detection power is defined as

$$\mathrm{P_D}(\mathcal{C}) = P_{H_1}\big(\text{declare hallucination} \mid \mathcal{C}\big),$$
$$\text{s.t. } \mathrm{P_F}(\mathcal{C}) = P_{H_0}\big(\text{declare hallucination} \mid \mathcal{C}\big) = \alpha.$$

Effectiveness of our method. Table 1 reports AUROC and detection power at a fixed 10% false-alarm rate across datasets and models (for HaluEval, we set $\theta = 0.2$). We emphasize that **our approach serves as a universal evaluation framework, delivers robust performance across diverse models and datasets, and achieves the best results in most settings**. Our method consistently achieves superior detection power in each evaluated dataset-model pair compared to all baselines. Regarding AUROC, our method improves upon the strongest baseline by at least 0.65% on CoQA and 0.22% on TriviaQA; on HaluEval, our performance is comparable to the best-performing baseline (either exceeding it by at least 0.77% or trailing it by at most 0.16%). Notably, relative to the worst-performing baseline, our method yields substantial gains in AUROC ranging from 8.5% to 28.3%.

We further study the effect of annotation protocols, comparing Rouge-L annotation with LLM-as-a-judge annotation. For the latter, we use Llama-3.1-8B-Instruct to score consistency between the reference answer and the generated answer; we deem an answer correct if its consistency score exceeds 80 out of 100 (denoted as Llama annotation). As shown in Figure 1, the performance of individual baselines can shift under different annotations: SE-based scores and the spectral eigenvalue score improve a little under Llama annotation, while lexical similarity degrades substantially because it is sensitive to surface overlap and aligns more naturally with Rouge-L. Despite these shifts, even when some baselines degrade, our method remains robust, staying close to the best score and substantially improving over the weakest baseline.

Figure 1 also includes two simple aggregation strategies, majority voting and averaging. Our aggregation outperforms them in most cases and remains competitive otherwise. Notably, under Llama annotation, neither majority voting nor averaging remains close to the best baseline due to

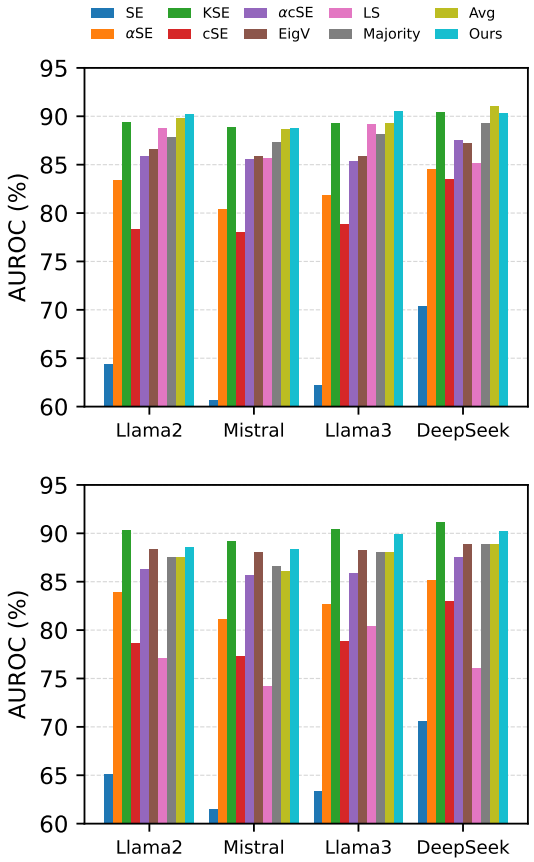

Figure 1: AUROC on the HaluEval dataset under Rouge-L annotation (top) and Llama annotation (bottom).

the influence of poorly performing scores, whereas our method consistently approaches the best score. This highlights the more consistently robust behavior of our approach.

To further validate robustness, Figure 2 aggregates results over all model–dataset pairs (with per-pair results reported in the appendix). Our method achieves the highest average AUROC among all individual baselines and the two aggregation baselines, supporting its robustness across settings.

Because our approach makes no prior assumptions about datasets or models, a particular metric may outperform our method in specific cases. However, such gains are typically small, and every baseline score deteriorates in some regimes. For example, SE, αSE, KSE, and EigV perform worse on GSM8K; clustered_SE and α_clustered_SE perform poorly on Qwen3; and LS drops substantially under Llama annotation, despite being among the strongest scores on GSM8K. These inconsistencies suggest that individual scores are unreliable when the test-time prompt distribution is unknown. In summary, our method effectively integrates the

Table 1: AUROC (%) and detection power (P_D, %) at 10% false alarm rate across different models and datasets. $*$ and † indicate the best and worst performance in each setting, respectively. Boldface highlights our method, which achieves the highest performance in the majority of evaluated cases.

Dataset	Method	Llama-2-13B		Mistral-7B		Llama-3.1-8B		DeepSeek-v2-Lite	
		AUROC	P_D	AUROC	P_D	AUROC	P_D	AUROC	P_D
HaluEval	SE	$64.34^{\dagger} \pm 0.23$	$41.50^{\dagger} \pm 0.58$	$60.69^{\dagger} \pm 0.23$	$35.68^{\dagger} \pm 0.57$	$62.24^{\dagger} \pm 0.23$	$36.88^{\dagger} \pm 0.58$	$70.36^{\dagger} \pm 0.23$	50.99 ± 0.38
	αSE	83.43 ± 0.29	66.46 ± 0.87	80.36 ± 0.20	60.95 ± 0.66	81.80 ± 0.23	62.97 ± 0.67	84.54 ± 0.21	68.84 ± 0.86
	KSE	89.41 ± 0.38	71.21 ± 1.70	$88.85^{*} \pm 0.26$	69.90 ± 1.07	89.32 ± 0.19	69.35 ± 0.75	$90.43^{*} \pm 0.18$	72.27 ± 1.06
	clustered_SE	78.32 ± 0.34	51.03 ± 0.84	78.04 ± 0.28	50.06 ± 0.45	78.82 ± 0.16	51.99 ± 0.35	83.49 ± 0.22	60.70 ± 0.67
	α_clustered_SE	85.83 ± 0.31	71.77 ± 0.76	85.53 ± 0.16	67.32 ± 0.51	85.40 ± 0.13	68.18 ± 0.46	87.48 ± 0.15	71.32 ± 0.45
	EigV	86.59 ± 0.39	71.02 ± 1.00	85.83 ± 0.20	64.44 ± 0.85	85.90 ± 0.22	65.39 ± 0.66	87.16 ± 0.18	67.49 ± 0.62
	LS	88.77 ± 0.48	64.81 ± 3.87	85.62 ± 0.34	45.74 ± 2.35	89.22 ± 0.33	64.34 ± 1.36	85.19 ± 0.40	$41.28^{\dagger} \pm 5.64$
	Ours	$\mathbf{90.18^{*} \pm 0.34}$	$\mathbf{75.70^{*} \pm 1.19}$	$\mathbf{88.71 \pm 0.24}$	$\mathbf{71.74^{*} \pm 0.74}$	$\mathbf{90.56^{*} \pm 0.20}$	$\mathbf{74.93^{*} \pm 0.63}$	$\mathbf{90.27 \pm 0.16}$	$\mathbf{76.79^{*} \pm 0.96}$
CoQA	SE	$66.68^{\dagger} \pm 0.14$	$38.93^{\dagger} \pm 0.30$	$65.41^{\dagger} \pm 0.26$	$36.62^{\dagger} \pm 0.63$	$65.32^{\dagger} \pm 0.18$	$37.11^{\dagger} \pm 0.48$	$69.05^{\dagger} \pm 0.25$	$41.98^{\dagger} \pm 0.58$
	αSE	85.19 ± 0.20	65.10 ± 0.82	85.15 ± 0.23	65.42 ± 1.06	84.17 ± 0.18	61.96 ± 0.32	86.55 ± 0.25	66.73 ± 0.62
	KSE	88.72 ± 0.17	70.50 ± 0.57	88.43 ± 0.28	68.59 ± 0.74	87.77 ± 0.18	66.96 ± 0.80	89.05 ± 0.26	69.53 ± 1.01
	clustered_SE	85.56 ± 0.25	57.05 ± 0.91	85.55 ± 0.31	58.23 ± 1.07	85.18 ± 0.21	58.20 ± 0.46	86.42 ± 0.29	61.03 ± 0.69
	α_clustered_SE	89.79 ± 0.15	72.48 ± 0.71	90.04 ± 0.21	73.62 ± 1.04	89.46 ± 0.13	71.89 ± 0.39	90.80 ± 0.20	74.78 ± 0.54
	EigV	90.03 ± 0.15	72.08 ± 0.62	90.28 ± 0.22	73.49 ± 0.72	89.79 ± 0.15	72.07 ± 0.70	91.06 ± 0.26	74.34 ± 0.94
	LS	88.36 ± 0.49	64.33 ± 2.66	89.28 ± 0.25	68.25 ± 0.72	87.87 ± 0.37	61.92 ± 1.58	89.83 ± 0.52	70.94 ± 1.02
	Ours	$\mathbf{90.85^{*} \pm 0.11}$	$\mathbf{74.90^{*} \pm 0.84}$	$\mathbf{91.23^{*} \pm 0.25}$	$\mathbf{75.81^{*} \pm 1.34}$	$\mathbf{90.44^{*} \pm 0.13}$	$\mathbf{74.23^{*} \pm 0.36}$	$\mathbf{91.74^{*} \pm 0.19}$	$\mathbf{76.45^{*} \pm 1.47}$
TriviaQA	SE	$82.52^{\dagger} \pm 0.13$	67.96 ± 0.35	$82.12^{\dagger} \pm 0.13$	67.01 ± 0.44	$84.67^{\dagger} \pm 0.17$	69.02 ± 0.36	$87.34^{\dagger} \pm 0.18$	75.43 ± 0.34
	αSE	90.75 ± 0.10	78.50 ± 0.67	90.97 ± 0.11	78.02 ± 0.43	91.09 ± 0.16	77.32 ± 0.46	93.06 ± 0.12	84.15 ± 0.35
	KSE	92.06 ± 0.11	78.83 ± 0.47	92.50 ± 0.08	80.02 ± 0.38	92.31 ± 0.19	79.75 ± 0.61	94.68 ± 0.12	86.01 ± 0.42
	clustered_SE	90.08 ± 0.08	77.04 ± 0.25	90.93 ± 0.14	78.39 ± 0.35	91.98 ± 0.08	81.38 ± 0.21	94.01 ± 0.14	84.00 ± 0.38
	α_clustered_SE	93.94 ± 0.06	84.91 ± 0.23	94.48 ± 0.08	86.18 ± 0.32	94.57 ± 0.09	86.51 ± 0.33	95.65 ± 0.10	88.77 ± 0.24
	EigV	93.85 ± 0.07	84.18 ± 0.31	94.60 ± 0.08	86.18 ± 0.41	94.26 ± 0.08	84.81 ± 0.28	95.45 ± 0.10	88.06 ± 0.22
	LS	86.16 ± 0.70	$30.14^{\dagger} \pm 19.93$	85.57 ± 0.75	$0.00^{\dagger} \pm 0.00$	88.65 ± 0.66	$58.28^{\dagger} \pm 4.08$	88.89 ± 0.91	$60.32^{\dagger} \pm 2.49$
	Ours	$\mathbf{94.30^{*} \pm 0.09}$	$\mathbf{85.80^{*} \pm 0.56}$	$\mathbf{94.82^{*} \pm 0.07}$	$\mathbf{87.12^{*} \pm 0.43}$	$\mathbf{94.78^{*} \pm 0.15}$	$\mathbf{86.52^{*} \pm 0.64}$	$\mathbf{95.87^{*} \pm 0.11}$	$\mathbf{89.80^{*} \pm 0.41}$

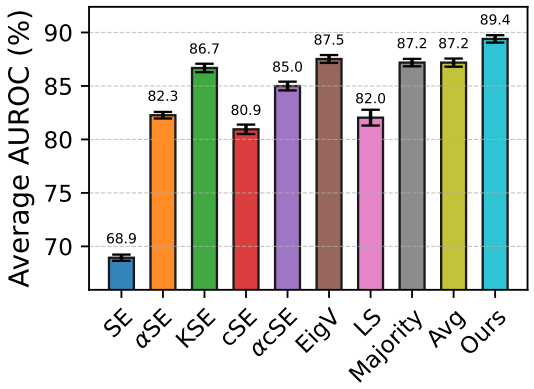

Figure 2: Average AUROC on all model-dataset pairs with Rouge-L annotation and Llama annotation.

strengths of different detection signals, yielding robust and consistent performance across models and datasets.

Ablation study. As reasoning models become increasingly common, we additionally evaluate newer models such as Qwen3-4B. We observe that semantic-entropy variants that rely only on cluster-assignment frequencies can degrade for long-form generations: long answers often differ in meaning to some extent, which makes clustering less informative and can weaken such scores. Because our method aggregates multiple scores, severely degraded baselines can in turn reduce overall performance. On CoQA, our method still outperforms all alternatives except EigV and LS.

We also study the effect of the calibration set size. In principle, increasing the calibration size reduces ϵ and may improve detection power. When we increase the calibration set from $1{,}000$ to $2{,}000$, the AUROC improvement is modest (up to 0.2%). Meanwhile, the standard deviation increases by more than 0.2%, likely because allocating more samples to calibration leaves fewer non-hallucinated examples for validating the false-alarm rate.

Next, we further enlarge the calibration set by merging CoQA and TriviaQA and sampling $3{,}000$ calibration prompts without hallucinated generations from the combined pool. This setting serves as a stress test for the realistic scenario where the test-time prompt distribution is unknown. Under this mixed calibration set, our performance changes only slightly, ranging from -0.21% to 0.27% on CoQA and TriviaQA.

In many LLM tasks, the available validation/test splits are relatively small, making it challenging to obtain a calibration set large enough to fully meet theoretical desiderata. Our results suggest that very large calibration sets are not required for strong empirical performance, and that mixing calibration data across distributions does not materially degrade performance.

Finally, the sensitivity of ROUGE-L annotations is analyzed by varying the tolerance threshold $\theta \in \{0.1, 0.2, 0.3\}$ and the ROUGE-L similarity threshold $\tau \in \{0.2, 0.3, 0.4\}$. As shown in Table 13 in the Appendix, increasing θ inherently increases task difficulty by introducing more noise into the non-hallucinated calibration set. While this

leads to a general decline in AUROC across all detectors, our proposed aggregation method remains the top or second-best performer in all settings. Notably, while base signals like Spectral Eigenvalue and Clustered SE exhibit high sensitivity to θ (dropping over 8% in AUROC on Mistral-7B), our framework maintains stability. This demonstrates that the multiple-testing procedure successfully leverages the most reliable signals available, even when other base detectors degrade. Furthermore, performance across all evaluated methods remained invariant to changes in τ, indicating that the framework is not overly sensitive to the specific similarity threshold used for ground-truth labeling.

5 Conclusions

In this work, we reconceptualize the hallucination detection problem as a hypothesis testing problem and introduce a multiple-testing-inspired approach to integrate various hallucination detection methods. Our method provides a theoretical guarantee of false-alarm control while empirically improving AUROC across different evaluation metrics. Notably, whereas existing methods can be inconsistent across metrics, our approach achieves superior or comparable performance across a broad range of models and datasets, showing robust generalization without requiring assumptions about the distribution of user queries posed to large language models (LLMs). Our proposed method has significant implications for the reliability and trustworthiness of LLM-generated content, particularly in critical applications such as healthcare, where misinformation can have severe consequences. By providing a robust hallucination detection mechanism with controllable false-alarm rates and negligible additional overhead beyond computing existing scores, our work supports safer deployment of LLM-based systems and reduces the risk that misleading or fabricated information is presented as fact.

6 Limitations

Despite its strengths, our method has a few limitations. Firstly, our evaluation relies on labeling prompts as inducing hallucinations or not; we currently use Rouge-L overlap or an LLM-based consistency judge. More capable LLM judges may better capture semantic equivalence under rephrasing and thus provide higher-quality ground-truth labels.

Secondly, as reasoning models show strong potential on complex tasks, further evaluation on such models, especially on more challenging settings such as complex mathematical reasoning or agentic problems, would be a valuable direction for future work.

Finally, our approach is aggregation-based and operates on a collection of base hallucination scores. In settings where one can propose multiple potentially reasonable candidate scores but it is unclear which will transfer best, this design is a feature: our procedure provides a principled way to combine them while controlling false alarms. That said, our method presumes access to at least one meaningful base score; for tasks where constructing such a score is itself challenging, score design remains a prerequisite and is complementary to our contribution.

Acknowledgment

This work was supported by the U.S. National Science Foundation (NSF) under grant 2106727. The authors also acknowledge the Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support (ACCESS) program (Boerner et al., 2023) for providing computing allocations and support through allocation CIS250917 (NSF grants #2138259, #2138286, #2138307, #2137603, and #2138296).

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Yoav Benjamini and Daniel Yekutieli. 2001. The control of the false discovery rate in multiple testing under dependency. *The Annals of Statistics*, 29(4):1165–1188.

Timothy J Boerner, Stephen Deems, Thomas R Furlani, Shelley L Knuth, and John Towns. 2023. Access: Advancing innovation: Nsfs advanced cyberinfrastructure coordination ecosystem: Services & support. In *Practice and experience in advanced research computing 2023: Computing for the common good*, pages 173–176.

Loredana Caruccio, Stefano Cirillo, Giuseppe Polese, Giandomenico Solimando, Shanmugam Sundaramurthy, and Genoveffa Tortora. 2024. Claude 2.0 large language model: Tackling a real-world classification problem with a new iterative prompt engineering approach. *Intelligent Systems with Applications*, 21:200336.

I Chern, Steffi Chern, Shiqi Chen, Weizhe Yuan, Kehua Feng, Chunting Zhou, Junxian He, Graham Neubig, Pengfei Liu, and 1 others. 2023. Factool: Factuality detection in generative ai–a tool augmented framework for multi-task and multi-domain scenarios. *arXiv preprint arXiv:2307.13528*.

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

Ignacio Meza De la Jara, Cristian Rodriguez-Opazo, Damien Teney, Damith Ranasinghe, and Ehsan Abbasnejad. 2025. Mysteries of the deep: Role of intermediate representations in out of distribution detection. *arXiv preprint arXiv:2510.05782*.

DeepSeek-AI. 2024. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. *Preprint*, arXiv:2405.04434.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. 2024. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630.

Marina Fomicheva, Shuo Sun, Lisa Yankovskaya, Frédéric Blain, Francisco Guzmán, Mark Fishel, Nikolaos Aletras, Vishrav Chaudhary, and Lucia Specia. 2020. Unsupervised quality estimation for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:539–555.

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*.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and 1 others. 2025. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55.

Siqing Huo, Negar Arabzadeh, and Charles LA Clarke. 2023. Retrieving supporting evidence for llms generated answers. *arXiv preprint arXiv:2306.13781*.

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

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

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

Ramneet Kaur, Colin Samplawski, Adam D Cobb, Anirban Roy, Brian Matejek, Manoj Acharya, Daniel Elenius, Alexander M Berenbeim, John A Pavlik, Nathaniel D Bastian, and 1 others. 2024. Addressing uncertainty in llms to enhance reliability in generative ai. *arXiv preprint arXiv:2411.02381*.

Nitish Shirish Keskar, Bryan McCann, Lav R Varshney, Caiming Xiong, and Richard Socher. 2019. Ctrl: A conditional transformer language model for controllable generation. *arXiv preprint arXiv:1909.05858*.

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

Faisal Ladhak, Esin Durmus, Mirac Suzgun, Tianyi Zhang, Dan Jurafsky, Kathleen McKeown, and Tatsunori B Hashimoto. 2023. When do pre-training biases propagate to downstream tasks? a case study in text summarization. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3206–3219.

Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. 2018. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in neural information processing systems*, 31.

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. *arXiv preprint arXiv:2305.11747*.

Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. 2023. Generating with confidence: Uncertainty quantification for black-box large language models. *arXiv preprint arXiv:2305.19187*.

Zheheng Luo, Qianqian Xie, and Sophia Ananiadou. 2023. Chatgpt as a factual inconsistency evaluator for text summarization. *arXiv preprint arXiv:2303.15621*.

Akshayaa Magesh, Venugopal V Veeravalli, Anirban Roy, and Susmit Jha. 2023. Principled out-of-distribution detection via multiple testing. *Journal of Machine Learning Research*, 24(378):1–35.

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

Diyana Muhammed, Gollam Rabby, and Sören Auer. 2025. Selfcheckagent: Zero-resource hallucination detection in generative large language models. *arXiv preprint arXiv:2502.01812*.

Alexander Nikitin, Jannik Kossen, Yarin Gal, and Pekka Marttinen. 2024. Kernel language entropy: Fine-grained uncertainty quantification for llms from semantic similarities. *Advances in Neural Information Processing Systems*, 37:8901–8929.

Yasumasa Onoe, Michael JQ Zhang, Eunsol Choi, and Greg Durrett. 2022. Entity cloze by date: What lms know about unseen entities. *arXiv preprint arXiv:2205.02832*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Miriam Rateike, Celia Cintas, John Wamburu, Tanya Akumu, and Skyler Speakman. 2023. Weakly supervised detection of hallucinations in llm activations. *arXiv preprint arXiv:2312.02798*.

Siva Reddy, Danqi Chen, and Christopher D Manning. 2019. Coqa: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266.

Mrinank Sharma, Meg Tong, Tomasz Korbak, David Duvenaud, Amanda Askell, Samuel R Bowman, Newton Cheng, Esin Durmus, Zac Hatfield-Dodds, Scott R Johnston, and 1 others. 2023. Towards understanding sycophancy in language models. *arXiv preprint arXiv:2310.13548*.

Karan Singhal, Tao Tu, Juraj Gottweis, Rory Sayres, Ellery Wulczyn, Mohamed Amin, Le Hou, Kevin Clark, Stephen R Pfohl, Heather Cole-Lewis, and 1 others. 2025. Toward expert-level medical question answering with large language models. *Nature Medicine*, pages 1–8.

Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, and 1 others. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and 1 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

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

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025a. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, Keming Lu, Mingfeng Xue, Runji Lin, Tianyu Liu, Xingzhang Ren, and Zhenru Zhang. 2024a. Qwen2.5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122*.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, and 1 others. 2024b. Qwen2. 5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122*.

Borui Yang, Md Afif Al Mamun, Jie M Zhang, and Gias Uddin. 2025b. Hallucination detection in large language models with metamorphic relations. *arXiv preprint arXiv:2502.15844*.

Jia-Yu Yao, Kun-Peng Ning, Zhen-Hui Liu, Mu-Nan Ning, Yu-Yang Liu, and Li Yuan. 2023. Llm lies: Hallucinations are not bugs, but features as adversarial examples. *arXiv preprint arXiv:2310.01469*.

Muru Zhang, Ofir Press, William Merrill, Alisa Liu, and Noah A Smith. 2023. How language model hallucinations can snowball. *arXiv preprint arXiv:2305.13534*.

Tianyi Zhang, Faisal Ladhak, Esin Durmus, Percy Liang, Kathleen McKeown, and Tatsunori B Hashimoto. 2024. Benchmarking large language models for news summarization. *Transactions of the Association for Computational Linguistics*, 12:39–57.

Chunting Zhou, Graham Neubig, Jiatao Gu, Mona Diab, Paco Guzman, Luke Zettlemoyer, and Marjan Ghazvininejad. 2020. Detecting hallucinated content in conditional neural sequence generation. *arXiv preprint arXiv:2011.02593*.

A Experiment Details

Table 2 shows the specific models used in our experiments. For each model, we generate 20 samples using pure sampling from the logits, with $\text{top\_k} = 0$, $\text{top\_p} = 1.0$, and $\text{temperature} = 1.0$. The hyperparameter α in Alpha Semantic Entropy score is fixed at 0.5. All experiments are conducted on a single L40S GPU, except for LLaMA-2-13B, which is evaluated using two L40S GPUs. Sampling generations and computing scores for each test take approximately three days, but the process can be accelerated through parallel computing.

The computational overhead is not expensive in our experiments. Retrieving token probabilities requires most of the time (several hours), and after that, different versions of semantic entropy computations are finished within one hour separately, spectral eigenvalue and lexical similarity could be finished within one hour separately too. Our multiple testing framework is finished within 0.1 second then.

We adopt the QA split of HaluEval with $10,000$ questions, the validation split of TriviaQA with $9,960$ questions, the development split of CoQA with $7,983$ questions, and the test split of GSM8K with 1319 questions. Since GSM8K is harder for LLMs to answer, θ was chosen to be 0.3, and the calibration dataset size is 500.

Table 2: Models tested in our experiments and their Hugging Face identifiers

Model	Hugging Face Identifier
Llama-2-13B	meta-llama/Llama-2-13b-hf
Mistral-7B	mistralai/Mistral-7B-v0.1
Llama-3.1-8B	meta-llama/Llama-3.1-8B
DeepSeek-v2-Lite	deepseek-ai/DeepSeek-V2-Lite
Qwen2.5-Math-1.5B-Instruct	Qwen/Qwen2.5-Math-1.5B-Instruct
Llama-3.2-3B-Instruct	meta-llama/Llama-3.2-3B-Instruct
Qwen3-4B	Qwen/Qwen3-4B

Table 3 reports the hallucination proportions (# hallucinated samples/# Total samples) derived from the experimental splits for each model-dataset pair. These statistics reveal that hallucination prevalence varies significantly across model architectures and annotation regimes. In many instances, such as within the HaluEval and TriviaQA benchmarks, hallucination rates exceed 70%, creating high-noise environments. The consistency of our results across these diverse proportions underscores the robustness of the proposed framework.

Table 3: Hallucination proportions (%) across benchmarks.

Dataset	Model	Hallucination Rate (%)	
		(Rouge-L Annotation)	(Llama Annotation)
HaluEval	Llama-3.1-8B	70.89	81.01
	DeepSeek-V2	71.97	80.61
	Llama-2-13B	79.11	85.97
	Mistral-7B	70.53	80.04
TriviaQA	Llama-3.1-8B	74.75	78.73
	Llama-2-13B	72.56	76.32
	Mistral-7B	73.53	76.46
	DeepSeek-V2	69.20	72.66
CoQA	Llama-3.1-8B	63.47	83.55
	Llama-2-13B	64.90	83.70
	DeepSeek-V2	72.59	88.36
	Mistral-7B	68.53	86.45
	Qwen3-4B	85.43	87.32
GSM8K	Qwen2.5-Math	77.71	66.26
	Llama-3.2-3B	55.88	47.61

Table 4: Detection power (%) at 10% false alarm rate across different models and datasets with a calibration data size of 3,000. $*$ and † indicate the best and the worst detection power in each case.

Dataset	Method	Llama-2-13B	Mistral-7B	Llama-3.1-8B	DeepSeek-v2-Lite
CoQA	SE	$38.82^{\dagger} \pm 0.71$	$36.65^{\dagger} \pm 0.66$	$37.42^{\dagger} \pm 0.64$	$41.86^{\dagger} \pm 0.56$
	αSE	65.54 ± 1.05	65.45 ± 1.41	62.19 ± 1.10	66.65 ± 0.88
	KSE	70.46 ± 0.92	68.06 ± 1.23	67.44 ± 0.93	69.94 ± 0.84
	clustered_SE	57.21 ± 1.29	58.50 ± 1.42	58.52 ± 0.98	60.79 ± 1.06
	α_clustered_SE	72.79 ± 0.71	73.13 ± 0.77	71.65 ± 0.89	74.64 ± 0.52
	EigV	71.80 ± 1.01	73.42 ± 0.80	72.20 ± 1.47	74.01 ± 1.30
	LS	64.84 ± 4.34	67.41 ± 1.20	62.95 ± 1.91	70.32 ± 1.27
	Ours	$\mathbf{74.48^{*} \pm 0.37}$	$\mathbf{74.03^{*} \pm 1.18}$	$\mathbf{73.55^{*} \pm 1.27}$	$\mathbf{76.36^{*} \pm 1.22}$
TriviaQA	SE	67.99 ± 0.44	66.52 ± 0.48	69.00 ± 0.71	75.30 ± 0.26
	αSE	78.43 ± 0.80	77.55 ± 0.50	77.38 ± 0.83	83.92 ± 0.50
	KSE	78.71 ± 0.58	79.78 ± 0.50	79.71 ± 0.91	85.67 ± 0.61
	clustered_SE	76.98 ± 0.19	78.27 ± 0.70	81.25 ± 0.71	84.04 ± 0.60
	α_clustered_SE	84.84 ± 0.37	86.05 ± 0.56	86.47 ± 0.64	88.60 ± 0.35
	EigV	83.95 ± 0.29	86.01 ± 0.41	84.69 ± 0.64	87.92 ± 0.36
	LS	$10.44^{\dagger} \pm 13.99$	$0.00^{\dagger} \pm 0.00$	$57.72^{\dagger} \pm 3.76$	$60.66^{\dagger} \pm 6.73$
	Ours	$\mathbf{86.52^{*} \pm 0.34}$	$\mathbf{87.07^{*} \pm 0.71}$	$\mathbf{87.82^{*} \pm 0.70}$	$\mathbf{89.93^{*} \pm 0.25}$
CoQA +TriviaQA	SE	54.96 ± 0.55	52.70 ± 0.36	$56.64^{\dagger} \pm 0.41$	$58.79^{\dagger} \pm 0.18$
	αSE	73.03 ± 0.54	72.65 ± 0.59	70.72 ± 0.52	76.73 ± 0.62
	KSE	74.93 ± 0.38	74.60 ± 0.77	74.38 ± 0.46	79.07 ± 0.76
	clustered_SE	67.20 ± 0.24	68.27 ± 0.46	68.99 ± 0.31	73.15 ± 0.38
	α_clustered_SE	79.67 ± 0.30	80.67 ± 0.52	80.42 ± 0.43	82.83 ± 0.36
	EigV	78.82 ± 0.30	80.78 ± 0.46	78.99 ± 0.51	82.18 ± 0.64
	LS	$50.14^{\dagger} \pm 1.68$	$51.85^{\dagger} \pm 2.87$	60.58 ± 2.71	59.93 ± 2.67
	Ours	$\mathbf{81.48^{*} \pm 0.34}$	$\mathbf{81.81^{*} \pm 0.55}$	$\mathbf{81.81^{*} \pm 0.62}$	$\mathbf{84.06^{*} \pm 0.49}$

B Additional Results

Calibration Dataset Expansion. Table 4 and Table 5 correspond to constructing a mixed calibration set by pooling non-hallucinated prompts from TriviaQA and CoQA. As noted in the main text, AUROC changes only slightly (within 0.3%), and detection power also varies modestly (within 1.8%). Table 6 and Table 7 consider a second strategy, where we sample $2,000$ non-hallucinated prompts and use the remaining non-hallucinated prompts to estimate the false-alarm rate. Performance remains comparable to the default setting with $|\mathcal{C}| = 1,000$. The standard deviation is larger, likely because the number of held-out non-hallucinated examples used to estimate the false-alarm rate is smaller in each case (ranging from 188 to $1,068$). Overall, these results further support the robustness of our method, which remains the top-performing approach across all settings.

Table 5: AUROC (%) across different models and datasets with a calibration data size of 3,000. $*$ and † indicate the best and worst AUROC in each case.

Dataset	Method	Llama-2-13B	Mistral-7B	Llama-3.1-8B	DeepSeek-v2-Lite
CoQA	SE	$66.76^{\dagger} \pm 0.27$	$65.46^{\dagger} \pm 0.38$	$65.55^{\dagger} \pm 0.31$	$68.99^{\dagger} \pm 0.31$
	αSE	85.21 ± 0.20	85.28 ± 0.38	84.29 ± 0.39	86.48 ± 0.21
	KSE	88.77 ± 0.30	88.53 ± 0.26	87.85 ± 0.28	89.03 ± 0.27
	clustered_SE	85.68 ± 0.25	85.65 ± 0.30	85.41 ± 0.37	86.33 ± 0.28
	α_clustered_SE	89.81 ± 0.12	90.08 ± 0.20	89.56 ± 0.26	90.73 ± 0.23
	EigV	89.98 ± 0.25	90.36 ± 0.29	89.85 ± 0.25	91.00 ± 0.31
	LS	88.49 ± 0.57	89.17 ± 0.40	88.01 ± 0.48	89.72 ± 0.49
	Ours	$\mathbf{90.82^{*} \pm 0.15}$	$\mathbf{91.02^{*} \pm 0.20}$	$\mathbf{90.52^{*} \pm 0.31}$	$\mathbf{91.57^{*} \pm 0.25}$
TriviaQA	SE	$82.57^{\dagger} \pm 0.21$	$82.00^{\dagger} \pm 0.20$	$84.77^{\dagger} \pm 0.36$	$87.31^{\dagger} \pm 0.21$
	αSE	90.68 ± 0.17	90.91 ± 0.21	91.16 ± 0.26	92.99 ± 0.24
	KSE	92.09 ± 0.16	92.46 ± 0.17	92.39 ± 0.28	94.63 ± 0.28
	clustered_SE	90.06 ± 0.08	90.87 ± 0.24	92.02 ± 0.26	94.01 ± 0.19
	α_clustered_SE	93.85 ± 0.07	94.46 ± 0.18	94.61 ± 0.21	95.62 ± 0.15
	EigV	93.77 ± 0.09	94.60 ± 0.14	94.28 ± 0.19	95.42 ± 0.19
	LS	85.11 ± 0.65	85.29 ± 1.09	88.40 ± 0.80	89.20 ± 0.84
	Ours	$\mathbf{94.44^{*} \pm 0.10}$	$\mathbf{94.81^{*} \pm 0.13}$	$\mathbf{95.05^{*} \pm 0.22}$	$\mathbf{95.94^{*} \pm 0.18}$
TriviaQA +CoQA	SE	$75.35^{\dagger} \pm 0.18$	$74.12^{\dagger} \pm 0.19$	$76.45^{\dagger} \pm 0.26$	$78.00^{\dagger} \pm 0.19$
	αSE	88.28 ± 0.18	88.37 ± 0.15	88.26 ± 0.22	90.09 ± 0.24
	KSE	90.62 ± 0.17	90.72 ± 0.17	90.42 ± 0.19	92.39 ± 0.27
	clustered_SE	87.56 ± 0.09	88.04 ± 0.16	88.50 ± 0.24	90.39 ± 0.18
	α_clustered_SE	92.00 ± 0.07	92.48 ± 0.14	92.40 ± 0.17	93.63 ± 0.14
	EigV	92.11 ± 0.13	92.77 ± 0.17	92.40 ± 0.17	93.64 ± 0.19
	LS	86.40 ± 0.60	86.32 ± 0.48	88.35 ± 0.39	88.79 ± 0.49
	Ours	$\mathbf{92.74^{*} \pm 0.11}$	$\mathbf{93.07^{*} \pm 0.11}$	$\mathbf{92.98^{*} \pm 0.17}$	$\mathbf{94.01^{*} \pm 0.17}$

Table 6: Detection power (%) at 10% false alarm rate across different models and datasets with a calibration data size of 2,000. $*$ and † indicate the best and the worst detection power in each case.

Dataset	Method	Llama-2-13B	Mistral-7B	Llama-3.1-8B	DeepSeek-v2-Lite
CoQA	SE	$38.69^{\dagger} \pm 0.81$	$37.55^{\dagger} \pm 1.44$	$37.11^{\dagger} \pm 0.93$	$42.73^{\dagger} \pm 2.12$
	αSE	65.97 ± 1.63	65.11 ± 2.79	62.30 ± 0.62	66.28 ± 4.25
	KSE	70.67 ± 1.28	68.89 ± 1.34	67.27 ± 1.02	70.07 ± 3.20
	clustered_SE	56.96 ± 2.20	59.03 ± 2.58	58.63 ± 1.01	59.31 ± 3.31
	α_clustered_SE	72.57 ± 1.35	73.64 ± 1.69	71.47 ± 1.09	74.32 ± 2.04
	EigV	72.54 ± 1.26	74.34 ± 1.49	72.01 ± 1.72	72.77 ± 3.01
	LS	62.34 ± 5.29	68.91 ± 2.71	60.91 ± 3.58	65.05 ± 8.28
	Ours	$\mathbf{74.72^{*} \pm 1.28}$	$\mathbf{76.53^{*} \pm 2.70}$	$\mathbf{74.69^{*} \pm 1.01}$	$\mathbf{75.81^{*} \pm 2.47}$
TriviaQA	SE	67.82 ± 0.68	67.12 ± 0.82	68.66 ± 1.63	75.48 ± 0.55
	αSE	78.44 ± 1.09	77.98 ± 1.19	77.46 ± 1.46	84.16 ± 0.55
	KSE	78.84 ± 1.39	80.22 ± 0.90	79.65 ± 1.16	85.87 ± 0.73
	clustered_SE	76.78 ± 1.26	78.40 ± 0.86	81.39 ± 1.26	83.91 ± 0.48
	α_clustered_SE	84.93 ± 0.77	86.21 ± 1.18	86.33 ± 1.07	88.75 ± 0.39
	EigV	84.22 ± 0.74	86.30 ± 1.00	84.90 ± 1.07	88.11 ± 0.49
	LS	$34.19^{\dagger} \pm 23.91$	$0.00^{\dagger} \pm 0.00$	$54.87^{\dagger} \pm 7.37$	$58.38^{\dagger} \pm 6.66$
	Ours	$\mathbf{85.79^{*} \pm 0.81}$	$\mathbf{87.28^{*} \pm 1.01}$	$\mathbf{86.52^{*} \pm 1.17}$	$\mathbf{89.81^{*} \pm 0.60}$

Results on math datasets. Because math questions are more challenging, we focus on instruction-tuned models, Qwen2.5-Math-1.5B-Instruct (Yang et al., 2024a) and LLaMA-3.2-3B-Instruct (Grattafiori et al., 2024), using chain-of-thought prompting with the chat-style template. Because our evaluation setting aims not to assume any particular prompt type, we do not post-process generations (e.g., extracting the final numeric answer), as is common in prior work on math benchmarks. Instead, we keep the full generations, consistent with our other QA datasets. For annotation, however, we extract the final numeric answer from each sampled generation when it exists.

As shown in Table 8 and Table 9, our method remains robust. Lexical similarity is the only baseline that achieves consistently strong performance, while several others degrade substantially (some even perform worse than random guessing). In the presence of these weak baselines, our method attains the second-best overall performance. These

Table 7: AUROC (%) across different models and datasets with a calibration data size of 2,000. $*$ and † indicate the best and worst AUROC in each case.

Dataset	Method	Llama-2-13B	Mistral-7B	Llama-3.1-8B	DeepSeek-v2-Lite
CoQA	SE	$66.59^{\dagger} \pm 0.49$	$65.76^{\dagger} \pm 0.61$	$65.35^{\dagger} \pm 0.31$	$69.38^{\dagger} \pm 1.29$
	αSE	85.38 ± 0.40	85.38 ± 0.57	84.17 ± 0.33	86.82 ± 0.64
	KSE	88.86 ± 0.27	88.53 ± 0.55	87.86 ± 0.26	89.46 ± 0.67
	clustered_SE	85.55 ± 0.70	85.64 ± 0.75	85.25 ± 0.27	86.46 ± 1.16
	α_clustered_SE	89.85 ± 0.40	90.09 ± 0.50	89.41 ± 0.31	90.94 ± 0.51
	EigV	90.10 ± 0.30	90.43 ± 0.59	89.80 ± 0.35	91.28 ± 0.69
	LS	88.16 ± 0.77	89.54 ± 0.80	87.87 ± 0.78	89.46 ± 1.41
	Ours	$\mathbf{90.90^{*} \pm 0.36}$	$\mathbf{91.44^{*} \pm 0.60}$	$\mathbf{90.54^{*} \pm 0.26}$	$\mathbf{91.70^{*} \pm 0.51}$
TriviaQA	SE	$82.44^{\dagger} \pm 0.45$	$82.19^{\dagger} \pm 0.50$	$84.77^{\dagger} \pm 0.60$	$87.36^{\dagger} \pm 0.22$
	αSE	90.72 ± 0.29	91.03 ± 0.48	91.24 ± 0.46	93.08 ± 0.20
	KSE	92.06 ± 0.26	92.68 ± 0.44	92.30 ± 0.40	94.73 ± 0.22
	clustered_SE	90.02 ± 0.30	90.98 ± 0.43	92.06 ± 0.53	93.99 ± 0.21
	α_clustered_SE	93.94 ± 0.22	94.56 ± 0.35	94.65 ± 0.40	95.66 ± 0.19
	EigV	93.80 ± 0.21	94.72 ± 0.33	94.30 ± 0.43	95.49 ± 0.16
	LS	86.43 ± 0.55	85.40 ± 1.29	88.26 ± 0.93	89.31 ± 0.56
	Ours	$\mathbf{94.29^{*} \pm 0.21}$	$\mathbf{94.90^{*} \pm 0.31}$	$\mathbf{94.81^{*} \pm 0.40}$	$\mathbf{95.92^{*} \pm 0.18}$

differences arise because, across samples, math solutions often differ in their intermediate reasoning, rephrasings, or decomposition of the problem, which can distort semantic-similarity signals and degrade semantic-entropy variants as well as spectral scores. In contrast, lexical similarity primarily captures surface overlap, and thus better reflects whether two generations are following a similar solution trajectory rather than diverging into irrelevant content or generic refusals. We also report majority voting and averaging; our method substantially improves over these aggregation strategies in most settings.

Table 8: Detection power (%) at 10% false alarm rate across different models on GSM8K. $*$ and † indicate the best and the worst detection power in each case.

Dataset	Method	Qwen2.5-Math-1.5B-Instruct	Llama-3.2-3B-Instruct
GSM8K	SE	$0.21^{\dagger} \pm 0.00$	$0.77^{\dagger} \pm 0.00$
	αSE	5.05 ± 0.00	3.23 ± 0.00
	KSE	8.84 ± 1.43	7.45 ± 1.54
	clustered_SE	16.59 ± 2.82	28.12 ± 2.52
	α_clustered_SE	12.75 ± 0.64	23.40 ± 3.92
	EigV	9.20 ± 1.93	6.02 ± 1.72
	LS	$32.66^{*} \pm 2.73$	$63.91^{*} \pm 3.94$
	Majority Voting	12.99 ± 3.47	16.05 ± 2.92
	Averaging	11.35 ± 2.56	35.26 ± 4.69
	Ours	$\mathbf{18.19 \pm 3.20}$	$\mathbf{44.23 \pm 4.83}$

Table 9: AUROC (%) across different models on GSM8K. $*$ and † indicate the best and worst AUROC in each case.

Dataset	Method	Qwen2.5-Math-1.5B-Instruct	Llama-3.2-3B-Instruct
GSM8K	SE	49.54 ± 0.17	50.21 ± 0.21
	αSE	48.87 ± 0.61	49.47 ± 0.75
	KSE	$47.99^{\dagger} \pm 1.45$	46.34 ± 1.90
	clustered_SE	61.24 ± 1.31	63.61 ± 1.77
	α_clustered_SE	60.00 ± 1.61	62.34 ± 2.01
	EigV	49.46 ± 1.47	$43.57^{\dagger} \pm 1.92$
	LS	$75.57^{*} \pm 1.06$	$83.79^{*} \pm 0.75$
	Majority Voting	54.26 ± 1.35	54.33 ± 1.84
	Averaging	64.70 ± 1.57	71.92 ± 1.50
	Ours	$\mathbf{62.24 \pm 1.39}$	$\mathbf{74.10 \pm 1.58}$

Results on the reasoning model. Table 10 reports results on the reasoning model Qwen3-4B on

Table 10: Detection power (%) at 10% false alarm rate and AUROC (%) on Qwen3-4B and CoQA. $*$ and † indicate the best and worst performance in each case.

Method	AUROC	Detection Power
SE	51.88 ± 0.40	9.15 ± 0.00
αSE	61.51 ± 0.51	20.70 ± 0.73
KSE	73.47 ± 0.50	39.03 ± 1.45
clustered_SE	$33.55^{\dagger} \pm 0.87$	1.45 ± 0.38
α_clustered_SE	36.55 ± 0.87	$1.39^{\dagger} \pm 0.26$
EigV	$81.88^{*} \pm 0.80$	56.36 ± 3.26
LS	80.87 ± 0.63	$58.06^{*} \pm 2.06$
Majority Voting	67.93 ± 0.48	35.24 ± 2.18
Averaging	47.14 ± 0.75	4.69 ± 0.51
Ours	$\mathbf{76.09 \pm 0.67}$	$\mathbf{40.70 \pm 2.35}$

CoQA. As noted in the main text, longer generations often differ slightly in meaning across samples, which weakens baselines that rely on discrete semantic clusters. This effect is most pronounced for the clustered variants, where performance can drop substantially, even below random guessing. Despite being influenced by these weak scores, our method remains robust: it stays close to the strongest baselines (EigV and LS) and outperforms the remaining methods.

Table 11: Detection power (%) at 10% false alarm rate across different models on HaluEval under different annotations. An asterisk ($*$) indicates the best detection power in each case, while a dagger (†) indicates the worst. Our proposed method demonstrates robust performance across all cases.

Annotation	Method	Llama-2-13B	Mistral-7B	Llama-3.1-8B	DeepSeek-v2-Lite
Llama	SE	41.70 ± 1.18	35.99 ± 0.51	$38.40^{\dagger} \pm 0.68$	49.60 ± 1.00
	αSE	67.96 ± 1.05	62.36 ± 0.57	65.56 ± 0.83	70.02 ± 0.79
	KSE	74.65 ± 1.56	71.43 ± 1.21	74.87 ± 0.73	76.43 ± 0.94
	clustered_SE	52.58 ± 1.61	50.64 ± 0.61	51.98 ± 0.76	61.10 ± 0.86
	α_clustered_SE	72.10 ± 1.22	68.67 ± 0.76	68.49 ± 0.49	71.36 ± 0.83
	EigV	75.16 ± 1.19	70.67 ± 0.85	70.24 ± 0.59	71.91 ± 0.49
	LS	$22.48^{\dagger} \pm 7.73$	$11.06^{\dagger} \pm 3.97$	39.37 ± 3.58	$0.00^{\dagger} \pm 0.00$
	Majority Voting	72.24 ± 1.64	68.55 ± 0.77	70.98 ± 0.88	72.76 ± 1.00
	Averaging	71.17 ± 1.45	65.95 ± 0.71	70.39 ± 0.80	71.89 ± 1.16
	Ours	$\mathbf{75.62^{*} \pm 1.27}$	$\mathbf{74.96^{*} \pm 1.00}$	$\mathbf{76.23^{*} \pm 0.80}$	$\mathbf{78.12^{*} \pm 1.04}$
RougeL	SE	$41.50^{\dagger} \pm 0.58$	$35.68^{\dagger} \pm 0.57$	$36.88^{\dagger} \pm 0.58$	50.99 ± 0.38
	αSE	66.46 ± 0.87	60.95 ± 0.66	62.97 ± 0.67	68.84 ± 0.86
	KSE	71.21 ± 1.70	69.90 ± 1.07	69.35 ± 0.75	72.27 ± 1.06
	clustered_SE	51.03 ± 0.84	50.06 ± 0.45	51.99 ± 0.35	60.70 ± 0.67
	α_clustered_SE	71.77 ± 0.76	67.32 ± 0.51	68.18 ± 0.46	71.32 ± 0.45
	EigV	71.02 ± 1.00	64.44 ± 0.85	65.39 ± 0.66	67.49 ± 0.62
	LS	64.81 ± 3.87	45.74 ± 2.35	64.34 ± 1.36	$41.28^{\dagger} \pm 5.64$
	Majority Voting	74.93 ± 0.75	70.31 ± 0.62	71.88 ± 0.32	75.41 ± 0.49
	Averaging	75.45 ± 0.67	69.77 ± 0.40	71.75 ± 0.68	76.71 ± 0.33
	Ours	$\mathbf{75.70^{*} \pm 1.19}$	$\mathbf{71.74^{*} \pm 0.74}$	$\mathbf{74.93^{*} \pm 0.63}$	$\mathbf{76.79^{*} \pm 0.96}$

Results under Rouge-L and Llama annotations. The results are reported in Table 11, Table 12, together with Table 1 and Figure 1. For CoQA with LLaMA annotations, we set the calibration set size to 800 because only a limited number of non-hallucinated prompts are annotated by Llama. Under Llama annotation, our method achieves the highest detection power on HaluEval (Table 11).

Table 12: AUROC (%) across different models and datasets under Llama annotation. $*$ and † indicate the best and worst AUROC in each case.

Dataset	Method	Llama-2-13B	Mistral-7B	Llama-3.1-8B	DeepSeek-v2-Lite
CoQA	SE	$65.85^{\dagger} \pm 0.36$	$65.78^{\dagger} \pm 0.54$	$64.07^{\dagger} \pm 0.35$	$69.62^{\dagger} \pm 1.00$
	αSE	85.12 ± 0.21	85.28 ± 0.30	83.15 ± 0.30	87.50 ± 0.89
	KSE	91.82 ± 0.38	91.94 ± 0.34	90.59 ± 0.33	92.36 ± 1.01
	clustered_SE	85.74 ± 0.72	86.83 ± 0.54	85.77 ± 0.28	88.53 ± 1.12
	α_clustered_SE	89.97 ± 0.41	90.61 ± 0.42	89.43 ± 0.30	92.22 ± 0.75
	EigV	$93.76^{*} \pm 0.27$	$94.23^{*} \pm 0.30$	$93.16^{*} \pm 0.31$	$94.75^{*} \pm 0.60$
	LS	73.53 ± 0.92	77.01 ± 1.26	73.56 ± 0.48	79.78 ± 2.60
	Majority Voting	91.11 ± 0.35	91.65 ± 0.32	90.30 ± 0.23	92.85 ± 0.83
	Averaging	88.73 ± 0.36	90.11 ± 0.33	88.24 ± 0.18	91.81 ± 0.89
	Ours	$\mathbf{91.91 \pm 0.31}$	$\mathbf{92.41 \pm 0.19}$	$\mathbf{91.11 \pm 0.19}$	$\mathbf{93.18 \pm 0.64}$
TriviaQA	SE	83.71 ± 0.09	83.23 ± 0.13	85.67 ± 0.17	88.08 ± 0.13
	αSE	91.96 ± 0.13	91.91 ± 0.15	92.25 ± 0.16	93.72 ± 0.11
	KSE	93.14 ± 0.14	93.49 ± 0.17	93.37 ± 0.18	95.64 ± 0.10
	clustered_SE	91.06 ± 0.14	91.93 ± 0.18	92.74 ± 0.15	94.53 ± 0.09
	α_clustered_SE	94.59 ± 0.08	95.24 ± 0.10	95.27 ± 0.13	96.06 ± 0.08
	EigV	$95.26^{*} \pm 0.08$	$95.88^{*} \pm 0.08$	$95.80^{*} \pm 0.09$	$96.58^{*} \pm 0.06$
	LS	$76.36^{\dagger} \pm 0.44$	$75.91^{\dagger} \pm 0.82$	$80.17^{\dagger} \pm 0.63$	$82.37^{\dagger} \pm 0.49$
	Majority Voting	94.56 ± 0.07	94.66 ± 0.11	95.11 ± 0.10	96.29 ± 0.08
	Averaging	93.71 ± 0.10	93.99 ± 0.10	94.98 ± 0.13	95.88 ± 0.06
	Ours	$\mathbf{94.94 \pm 0.08}$	$\mathbf{95.60 \pm 0.06}$	$\mathbf{95.53 \pm 0.11}$	$\mathbf{96.56 \pm 0.10}$

For AUROC on CoQA and TriviaQA under Llama annotation (Table 12), although lexical similarity degrades substantially and can affect aggregation, our method still ranks second and remains close to the best baseline (Spectral Eigenvalue), outperforming the remaining scores. A similar pattern holds on HaluEval: under Llama annotation (Figure 1), our method is again second and comparable to the best score (Kernel Semantic Entropy). Overall, these results further support the robustness of our approach: no single baseline performs well across all annotation protocols, whereas our method is consistently best or near-best.

Table 13: AUROC (%) on HaluEval with ROUGE-L annotation across varying θ (Fixed $\tau = 0.3$). Under each model, columns represent $\theta = 0.1/0.2/0.3$. $*$ and † indicate the best and worst AUROC in each setting.

Method	Llama-2-13B			Mistral-7B			Llama-3.1-8B			DeepSeek-V2		
	0.1	0.2	0.3	0.1	0.2	0.3	0.1	0.2	0.3	0.1	0.2	0.3
SE	NA	$64.34^{\dagger}$	$62.94^{\dagger}$	$64.06^{\dagger}$	$60.69^{\dagger}$	$58.81^{\dagger}$	$65.69^{\dagger}$	$62.24^{\dagger}$	$60.40^{\dagger}$	$73.59^{\dagger}$	$70.36^{\dagger}$	$67.59^{\dagger}$
αSE	NA	83.43	82.09	82.60	80.36	78.89	83.53	81.80	80.61	86.54	84.54	82.59
KSE	NA	89.41	88.13^{*}	89.79	88.85^{*}	87.95^{*}	90.05	89.32	88.79^{*}	91.65	90.43^{*}	88.60^{*}
clustered_SE	NA	78.32	76.03	82.30	78.04	75.59	82.75	78.82	76.96	87.74	83.49	80.40
α_clustered_SE	NA	85.83	82.85	89.71	85.53	82.69	89.39	85.40	82.47	91.46	87.48	84.47
EigV	NA	86.59	83.23	90.66	85.83	81.93	90.82	85.90	81.69	91.98	87.16	83.29
LS	NA	88.77	87.94	87.43	85.62	85.20	90.51	89.23	87.69	88.45	85.19	83.06
Ours	NA	$\mathbf{90.18^{*}}$	$\mathbf{87.79}$	$\mathbf{91.83^{*}}$	$\mathbf{88.71}$	$\mathbf{86.67}$	$\mathbf{92.67^{*}}$	$\mathbf{90.56^{*}}$	$\mathbf{88.13}$	$\mathbf{93.13^{*}}$	$\mathbf{90.27}$	$\mathbf{87.31}$

Note: NA indicates that the model did not produce enough consistent generations to meet the 1000-sample calibration requirement at $\theta = 0.1$.