
From Out-of-Distribution Detection to Hallucination Detection: A Geometric View

Anonymous Authors¹

Abstract

Detecting hallucinations in large language models is a critical open problem with significant implications for safety and reliability. While existing hallucination detection methods achieve strong performance in question-answering tasks, they remain less effective on tasks requiring reasoning. In this work, we revisit hallucination detection through the lens of out-of-distribution (OOD) detection, a well-studied problem in areas like computer vision. Treating next-token prediction in language models as a classification task allows us to apply OOD techniques, if we bring to bear appropriate modifications to account for the structural differences in large language models. We show that approaches based on OOD detection yield training-free, single-sample based detectors, achieving strong accuracy in hallucination detection in reasoning tasks. Overall, our work suggests that reframing hallucination detection as OOD detection provides a promising and scalable pathway toward language model safety.

1. Introduction

Hallucination remains a fundamental obstacle to the reliable deployment of Large Language Models (LLMs), and their detection has therefore been an important research area in recent years. One prominent line of work (Azaria & Mitchell; Kossen et al., 2024; Li et al., 2023; Liu et al.; Manakul et al., 2023; Marks & Tegmark; Su et al., 2024; Wang et al., 2024) trains a separate classifier to detect hallucinations. These methods can be sensitive to distribution shifts and incur high training costs. Training-free methods (Azaria & Mitchell; Kossen et al., 2024; Li et al., 2023; Liu et al.; Manakul et al., 2023; Marks & Tegmark; Su et al., 2024; Wang et al., 2024), by contrast, measure discrepancies across multiple

samples to detect hallucinations. While they avoid training overhead, they impose significant computational costs during inference—particularly in multi-step reasoning. Furthermore, although such methods perform well on concise question-answering tasks, they face challenges in reasoning, where the inherent diversity of valid reasoning paths makes comparing multiple outputs conceptually difficult. This motivates the need for *training-free, single-sample* based hallucination detection algorithms within the increasingly critical domain of reasoning.

Conceptually, hallucination detection is reminiscent of out-of-distribution (OOD) detection (Hendrycks et al., 2019; Liang et al., 2018; Liu et al., 2020; Sun et al., 2021; Sun & Li, 2022; Xu et al., 2024; Lee et al., 2018; Sun et al., 2022; Liu & Qin, 2024; 2025), a dominant research area for classification tasks. In OOD detection, the goal is to identify test samples whose class labels were not seen during training. For such samples, the model will still produce a prediction from the set of training classes, despite having no knowledge of the true label—effectively generating a “hallucination” of the classifier. At their core, both OOD detection and hallucination detection in LLMs boil down to measuring the model’s uncertainty (Kuhn et al., 2023b; Liu & Qin, 2024).

A natural question therefore is: can we leverage the rich literature on OOD detection to address the challenge of hallucination detection? Unfortunately, despite the conceptual similarities, several practical challenges arise when applying OOD detection methods to hallucination detection in LLMs. First, many OOD detectors rely on the empirical estimation of training statistics (Wang et al., 2022; Liu & Qin, 2025; Hendrycks et al., 2022; Lee et al., 2018; Liu et al., 2023) such as the mean of penultimate-layer features. While these statistics effectively characterize the training distribution and enable OOD detection, directly estimating them from LLM training data is impractical. Modern training corpora are often proprietary, and even when accessible, their scale renders computation prohibitively expensive. Second, the label space expands dramatically when moving from conventional classifiers to LLMs. OOD scores designed for small label spaces can become noisy and less reliable, and the expanded space furthermore amplifies computational costs, particularly in multi-step reasoning tasks where expenses

¹Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

055
056
057
058
059
060
061
062
063
064
065
066
067
068
069
070
071
072
073
074
075
076
077
078
079
080
081
082
083
084
085
086
087
088
089
090
091
092
093
094
095
096
097
098
099
100
101
102
103
104
105
106
107
108
109

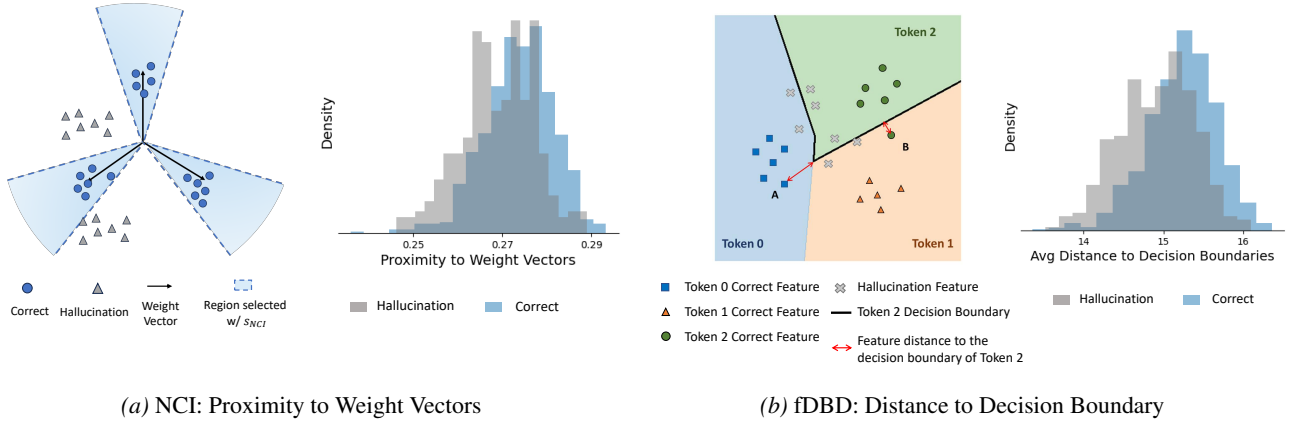


Figure 1. OOD-inspired geometric uncertainty measures can detect hallucinations. (a) Embeddings from hallucinated responses exhibit less proximity to weight vectors, extending OOD detector NCI (Liu & Qin, 2025). (b) Embeddings from hallucinated responses exhibit smaller distance to decision boundaries than correct embeddings, extending OOD detector fDBD (Liu & Qin, 2024). (a) Left and (b) Left illustrate the proximity score and distance to the decision boundaries defined in Definition 3.1 and Definition 3.2, respectively. (a) Right and (b) Right show histograms for the corresponding uncertainty measures based on the CSQA dataset on Llama-3.2-3B-Instruct.

accumulate across decoding steps. Third, unlike classifiers, which typically make deterministic decisions, LLMs are generative and may employ stochastic decoding. The inherent variance from token sampling therefore makes it unclear to what degree OOD scores are applicable in hallucination detection.

In this work, we take a step towards bridging the gap between OOD detection and hallucination detection. Since the cumulative computational cost in applying the score at each decoding step is high, we explore the adaptation of two light-weight OOD detectors, NCI (Liu & Qin, 2025) and fDBD (Liu & Qin, 2024). Both methods take a geometric view of measuring uncertainty: NCI measures penultimate layer feature proximity to weight vectors of the last layer (see Figure 1a) (the lower the proximity the higher the uncertainty); fDBD measures the penultimate layer distance to the decision boundaries of the last layer (Figure 1b). To adapt these methods to solve the task of hallucination detection, we derive an analytical proxy for the required training statistics in their score function - the mean of training features in this case. In addition, for fDBD to be efficient and effective in the large label space, we explore the reduction of the distance computation to a selected subset of high-likely alternative tokens in hallucination detection.

We furthermore show that a simple but effective way to extend these methods to sequences, is the averaging of the step-wise uncertainty score as the hallucination detection score. We demonstrate that the adapted geometry based methods can perform well under both greedy and stochastic decoding.

We perform extensive experiments to validate the effectiveness of OOD-inspired hallucination detection across different types of reasoning tasks (commonsense v.s. mathe-

tical), model architectures and model scales (Llama-3.2-3B-Instruct, Qwen-2.5-7B-Instruct, Qwen-3-32B), showing consistent superior performance compared to a range of baselines. This shows that framing hallucination detection as OOD detection provides a promising and scalable path toward improved language model safety.

2. Problem Statement

Let f denote a large language model (LLM). Given an input prompt x , let $y = (y_1, \dots, y_T)$ be the sequence of tokens generated by the model. We denote the set of all factually correct responses to x as $\mathcal{Y}^*(x)$. We say that the model f is *hallucinating* if the generated response y is incorrect, i.e., $y \notin \mathcal{Y}^*(x)$.

During inference, the ground-truth set of correct responses $\mathcal{Y}^*(x)$ for a given prompt x is unknown. The task of hallucination detection is to construct a **binary detector** \mathcal{I} that predicts whether $y \in \mathcal{Y}^*(x)$ based on the prompt x , the response y , and the model f 's internal representations:

$$\mathcal{I}(x, y; f) = \begin{cases} 1 \text{ (Hallucination)}, & \text{if } S(x, y; f) \leq \tau, \\ 0 \text{ (Not Hallucination)}, & \text{otherwise.} \end{cases} \quad (1)$$

where $S(x, y; f)$ is a scoring function and τ is a threshold. Under this formulation, a lower score S typically reflects lower model certainty and therefore a higher likelihood of hallucination.

3. Geometry of Token Generation under a Classification View

In this section, we adapt two geometric measures—feature proximity to weight vectors and feature distance to decision

boundaries—from the out-of-distribution detection literature for token generation. Both measures characterize the interplay between features and language head but were shown to exhibit different empirical strength across different OOD settings. We then discuss and empirically validate their connection to hallucination detection.

3.1. Classification View

Consider a large language model (LLM) f with model dimension d_{model} and vocabulary \mathcal{V} . At decoding step t , given an input prompt x and previously generated tokens $\mathbf{y}_{<t} = (\mathbf{y}_1, \dots, \mathbf{y}_{t-1})$, the model produces a penultimate-layer embedding $\mathbf{z}_x^t \in \mathbb{R}^{d_{\text{model}}}$. This representation is mapped by the language head f_{head} which generates logits over the vocabulary \mathcal{V} .

Conceptually, the language head f_{head} can be viewed as a high-dimensional linear classifier over the vocabulary \mathcal{V} , whose parameters are shared across decoding steps. Since f_{head} is linear, given embedding \mathbf{z} , during greedy decoding the next token to be generated is

$$\hat{c} = \arg \max_{v \in \mathcal{V}} \mathbf{w}_v^\top \mathbf{z} + b_v.$$

Here, \mathbf{w}_v and b_v denote the weight vector and bias associated with token v . The next token may differ from this definition under stochastic (non-greedy) decoding, as discussed in 4.4.

3.2. Feature Proximity to Weight Vectors

For OOD detection, Liu & Qin (2025) propose measuring model uncertainty based on the proximity of features to class weight vectors, introducing a Neural Collapse-inspired detector termed NCI. Following their framework, we define the proximity of a feature \mathbf{z} to the weight vectors of an LLM f as the scalar projection of the weight vector $\mathbf{w}_{\hat{c}}$ onto the centered feature $\mathbf{z} - \boldsymbol{\mu}_G$, where \hat{c} corresponds to the most confident token. Here, $\boldsymbol{\mu}_G$ denotes the estimated mean of training features; details on its estimation in LLM are discussed in Section 4.2.

Definition 3.1 (Feature Proximity to Weight Vectors). To adapt OOD detection setting based on NCI (Liu & Qin, 2025) to hallucination detection, we define the score s_{NCI} measuring the proximity of a feature \mathbf{z} on LLM f as:

$$s_{\text{NCI}}(\mathbf{z}) = \cos(\mathbf{w}_{\hat{c}}, \mathbf{z} - \boldsymbol{\mu}_G) \|\mathbf{w}_{\hat{c}}\|_2, \quad (2)$$

$$\text{where } \cos(\mathbf{w}_c, \mathbf{z} - \boldsymbol{\mu}_G) = \frac{(\mathbf{z} - \boldsymbol{\mu}_G) \cdot \mathbf{w}_c}{\|\mathbf{z} - \boldsymbol{\mu}_G\|_2 \|\mathbf{w}_c\|_2}.$$

A higher s_{NCI} indicates closer proximity to the weight vectors (see Figure 1a Left), corresponding to lower uncertainty. We remark that computing s_{NCI} per step requires $O(d_{\text{model}})$ operations.

3.3. Feature Distance to Decision Boundaries

As an alternative to NCI, Liu & Qin (2023) propose measuring model uncertainty based on feature distance to decision boundaries and introduce a fast decision-boundary-based OOD detector (fDBD).

In the following, we also explore adapting fDBD to hallucination detection, which can show stronger performance in some setting as we shall show. To this end, we define the region in the space of penultimate-layer representations, in which a token $c \in \mathcal{V}$ attains the maximum logit, as

$$\mathcal{R}_c = \left\{ \mathbf{z} \in \mathbb{R}^{d_{\text{model}}} \mid \arg \max_{v \in \mathcal{V}} \mathbf{w}_v^\top \mathbf{z} + b_v = c \right\}.$$

We refer to this region as the *decision region* of token c , and to its corresponding boundary as the *decision boundary* of token c , by analogy with multi-class classification. Under greedy decoding, this terminology is exact; we retain the same nomenclature when stochastic decoding is employed.

The distance from an embedding \mathbf{z} to the decision boundary of an token $c \neq \arg \max_{v \in \mathcal{V}} \mathbf{w}_v^\top \mathbf{z} + b_v$ is:

Definition 3.2 (Distance to Decision Boundary). Adapted from (Liu & Qin, 2024).

$$D_f(\mathbf{z}, c) = \inf_{\mathbf{z}' \in \mathcal{R}_c} \|\mathbf{z} - \mathbf{z}'\|_2.$$

This distance, illustrated in Figure 1b Left, corresponds to the minimal perturbation required to change the model’s most confident token from $\hat{c} = \arg \max_{v \in \mathcal{V}} \mathbf{w}_v^\top \mathbf{z} + b_v$ to $c \in \mathcal{V}, c \neq \hat{c}$. A larger distance corresponds to lower uncertainty.

We exclude token \hat{c} , the most confident token, from Definition 3.2, as the embedding \mathbf{z} already resides in the decision region of \hat{c} (and the distance is zero), carrying no uncertainty information. For other tokens $c \neq \hat{c}$, to avoid iteratively solving the optimization to measure the distance, we derive an efficient approximation by relaxing the decision region.

Theorem 3.3 (Approximate Distance to Decision Boundary). Adapted from (Liu & Qin, 2024). Given embedding \mathbf{z} and token $c \in \mathcal{V}, c \neq \arg \max_{v \in \mathcal{V}} \mathbf{w}_v^\top \mathbf{z} + b_v$, $D_f(\mathbf{z}, c)$ is lower bounded by

$$\tilde{D}_f(\mathbf{z}, c) := \frac{|(\mathbf{w}_{\hat{c}} - \mathbf{w}_c)^\top \mathbf{z} + (b_{\hat{c}} - b_c)|}{\|\mathbf{w}_{\hat{c}} - \mathbf{w}_c\|_2}. \quad (3)$$

See proof in Appendix C.

For each token c , computing the distance requires $O(d_{\text{model}})$ operations: $O(1)$ for the logit difference in the numerator and $O(d_{\text{model}})$ for the norm computation in the denominator.

3.4. Geometric Uncertainty Signals Hallucination

We empirically validate these two uncertainty measures in the context of hallucination detection in Figure 1.

As shown in Figure 1a, embeddings from hallucinated responses exhibit lower proximity to their corresponding weight vectors compared to correct responses, validating the effectiveness of NCI for hallucination detection. Furthermore, Figure 1b shows that penultimate-layer embeddings from hallucinated responses lie closer to decision boundaries than those from correct answers. This observation suggests that the uncertainty signal derived from the fDBD detector remains a robust indicator of hallucination.

Together, these results suggest that uncertainty measures from OOD detection transfer well to the hallucination detection setting. We next discuss the design of hallucination detectors that leverage and adapt these geometric properties.

4. From OOD to Hallucination Detection

While both OOD detection and hallucination detection are fundamentally linked to model uncertainty, the two problems differ significantly in scale and mechanism. In this section, we describe how we bridge these challenges and adapt methods originally developed for OOD detection to the context of hallucination detection.

4.1. Case Study Setups

In this case study, we examine hallucination detection on CSQA (Talmor et al., 2019) using Llama-3.2-3B-Instruct (Grattafiori et al., 2024). CSQA evaluates commonsense reasoning in a multiple-choice format, and we report results on the validation set, which contains 1,319 questions. We prompt the model using in-context learning examples following Wei et al. (2022). Details of the prompting strategy and answer extraction procedure are provided in Appendix B. Unless otherwise specified (Section 4.4), we use greedy decoding (i.e., $temp = 0$). Following Chen et al. (2024); Kuhn et al. (2023a); Lin et al. (2023; 2022); Malinin & Gales (2020), we quantify hallucination detection performance using the threshold-free metric area under the receiver operating characteristic curve (AUROC), where higher values indicate better performance. As a baseline, we consider the model’s Perplexity (Ren et al., 2023), defined as

$$\text{PPL}(\mathbf{y} | \mathbf{x}) = \exp \left(-\frac{1}{T} \sum_{t=1}^T \log p(\mathbf{y}_t | \mathbf{x}, \mathbf{y}_{<t}) \right).$$

Lower perplexity corresponds to higher model confidence in the generated output. To ensure consistency with Equation 1, we negate the perplexity score as the detection score.

4.2. Challenge I: Estimating Training Statistics at Scale

In OOD detection, many methods rely on statistics estimated from training features, such as the empirical mean or low-rank structure (e.g., SVD). For large language models,

however, computing such statistics over the full training corpus is infeasible due to its sheer scale. Even estimating them from a sampled subset is challenging: the training data are extremely diverse, and a limited subset can easily introduce bias into the estimation. As a result, purely data-driven estimates of training statistics become neither practical nor reliable in this setting. This motivates the need for *analytical, model-intrinsic estimations* of training statistics that do not require access to the training data.

In the following, we take the estimation of the training feature mean μ_G in the NCI score (Definition 3.1) as a step toward bridging this gap. Since the language head f_{head} is trained against the penultimate representations z , its parameters implicitly reflect the geometry of the training feature distribution. Specifically, the geometry of well-trained classifiers underlying NCI suggests that μ_G lies near the maximal-uncertainty point. We thus adopt this point as an analytical proxy for μ_G in the computation of NCI. Formally defining the proxy point as the *Decision-Neutral Closest Point*, obtained by minimizing the variance of logits across the vocabulary.

Lemma 4.1 (Analytical Solution for Decision-Neutral Closest Point). *Let $W \in \mathbb{R}^{|\mathcal{V}| \times d_{\text{model}}}$ be the weight matrix whose rows are w_v^\top , and let $\mathbf{b} \in \mathbb{R}^{|\mathcal{V}|}$ be the vector of biases b_v . The point z^* that minimizes logit variance across \mathcal{V} is given by:*

$$\hat{z}_* = -(W^\top P W)^\dagger W^\top P \mathbf{b} \tag{4}$$

where $P = I - \frac{1}{|\mathcal{V}|} \mathbf{1}\mathbf{1}^\top$ with I denoting the identity matrix, $\mathbf{1}$ the all-ones vector, and $(\cdot)^\dagger$ the Moore–Penrose pseudo-inverse.

The formal definition of the optimization objective of Decision-Neutral Point and the detailed derivation are provided in Appendix D.

We empirically evaluate the effectiveness of this analytical proxy for hallucination detection on CSQA using Llama-3.2-3B-Instruct (Table 1). Notably, the model employs a zero-bias language head ($\mathbf{b} = \mathbf{0} \in \mathbb{R}^{|\mathcal{V}|}$), implying that the analytical estimate of the mean by Equation 4 reduces to the origin of the embedding space, $\mathbf{0} \in \mathbb{R}^{d_{\text{model}}}$. For comparison, we also consider an empirical estimate of the training feature mean computed from the CSQA training set. Both the analytical proxy and the empirical estimate are incorporated into the step-wise NCI score (Equation 2). The final hallucination decision obtained by thresholding the average score across an output sequence of length T , i.e.,

$$S_{\text{NCI}} = \frac{1}{T} \sum_{t=0}^{T-1} s_{\text{NCI}}(z^t).$$

We further compare these variants against the Perplexity baseline, with results reported in Table 1.

Table 1. Analytical Proxy of Feature Mean Enables Effect Hallucination Detection. NCI with the analytical proxy outperforms the empirical estimation variant and Perplexity. AUROC reported for CSQA on Llama-3.2-3B-Instruct, with higher values indicating better performance; the best result is shown in bold.

Method	AUROC
Perplexity	63.23
NCI w/ empirical estimation	62.79
NCI w/ analytical proxy	66.07

As shown in Table 1, incorporating the analytical proxy of μ_G into the NCI score enables effective hallucination detection, outperforming both the baseline Perplexity and the NCI variant using the empirical mean. This demonstrates the effectiveness of analytically estimating training statistics for hallucination detection.

4.3. Challenge II: Effectiveness and Efficiency in Massive Vocabulary Space

Another challenge in extending OOD detectors to hallucination detection lies in the dramatically increased label space. While conventional classifiers typically involve on the order of thousands of classes at most, large language models operate over vocabularies containing hundreds of thousands of tokens. This scale poses challenges for directly applying OOD detectors such as fDBD, which indiscriminately aggregates signals from the entire label space.

Specifically, a direct adaptation of the fDBD detector to hallucination detection would compute the normalized average distance to the decision boundary over all tokens in the vocabulary:

$$s_{\text{fDBD}} := \frac{1}{|\mathcal{V}| - 1} \sum_{c \in \mathcal{V}, c \neq \hat{c}} \frac{\tilde{D}_f(z^t, c)}{\|z^t - \mu_G\|_2}, \quad (5)$$

where $\tilde{D}_f(z^t, c)$ is as defined in Definition 3.2. This formulation raises both effectiveness and efficiency concerns in the context of large vocabularies.

Intuitively, distances to the boundaries of low-probability tokens—such as irrelevant symbols, rare tokens, or numbers during plain-text reasoning—tend to be consistently large. These distant boundaries may offer less informative uncertainty signals and risk diluting the more meaningful distances associated with semantically plausible alternative tokens. Furthermore, Equation 5 scales linearly with vocabulary size, taking $O(d_{\text{model}}|\mathcal{V}|)$ computes per step; while feasible, such overhead is less than ideal.

Motivated by these, we restrict the distance computation to a small set of confident alternative tokens. Specifically, we select k tokens with the highest logits at step t , excluding the top-ranked token \hat{c} (which is omitted as the feature z^t already resides within its decision region). And we compute

Table 2. Controlling the alternative set size k improves performance of fDBD. AUROC on CSQA using Llama-3.2-3B-Instruct (higher is better). The best-performing k and the corresponding AUROC are highlighted in bold. While the method is effective across all choices of k , peak performance is achieved at $k = 1\,000$.

k	AUROC
1	68.64
10	68.76
100	69.18
1 000	69.24
10 000	68.87
100 000	68.26
All	68.15

the average normalized distance to the decision boundaries only with in the top- k set \mathcal{K}_t :

$$s_{\text{fDBD}}^k := \frac{1}{k} \sum_{c \in \mathcal{K}_t} \frac{\tilde{D}_f(z^t, c)}{\|z^t - \mu_G\|_2}. \quad (6)$$

This discards the potentially noisy uncertainty scores from low-probability tokens, while reducing the expected complexity at each step from $O(d_{\text{model}}|\mathcal{V}|)$ to $O(d_{\text{model}}k + |\mathcal{V}|)$ ¹. In practice, k can be chosen from the validation set as discussed in Section 5.

Empirically, controlling alternative set size gives benefit, as shown with CSQA on Llama-3.2-3B in Table 2. In these experiments, we use the analytical estimate z^* for μ_G as described in Section 4.2 and detect hallucination by thresholding the average score across T decoding steps for the output sequence of length T ,

$$S_{\text{fDBD}}^k = \frac{1}{T} \sum_{t=1}^T s_{\text{fDBD}}^k(z^t), \quad (7)$$

Specifically, we consider $k = \text{All}$, which includes all alternative tokens, as well as $k \in \{1, 10, 100, 1\,000, 10\,000, 100\,000\}$. As measured by AUROC (higher is better), the method remains effective across all choices of k consistently outperforming the Perplexity baseline (AUROC = 63.23). Performance peaks at $k = 1\,000$, suggesting that appropriately controlling the size of the alternative token set yields additional performance gains.

4.4. Challenge III: Robustness to Stochastic Generation

While conventional classifiers produce deterministic predictions, large language models can generate outputs either deterministically (e.g., via greedy decoding) or stochastically. So far, our experiments focus on greedy decoding,

¹Quickselect (Hoare, 1961) enables selecting the top- k tokens (unordered) in $O(|\mathcal{V}|)$ expected time.

Table 3. OOD-inspired hallucination detectors, NCI and fDBD, retain strong performance under stochastic decoding, significantly outperforming the baseline Perplexity. Results on CSQA using Llama-3.2-3B-Instruct. AUROC is reported (higher is better). For each temperature, mean \pm standard deviation is computed over five random seeds.

Method	temp = 0.2	temp = 0.5	temp = 0.8	temp = 1.0
Perplexity	63.49 \pm 0.75	62.08 \pm 2.34	63.45 \pm 0.71	62.68 \pm 1.08
NCI	67.07 \pm 0.53	66.04 \pm 1.59	67.53 \pm 0.88	67.93 \pm 1.65
fDBD	69.30 \pm 0.56	68.19 \pm 1.47	69.12 \pm 0.56	69.19 \pm 1.98

aligning with the deterministic setting of conventional classification problems. Under stochastic decoding, however, the generated token at each step may differ from the token with the highest logit. This poses a potential challenge for our hallucination detectors NCI and fDBD, as both are adapted from OOD detectors and are defined with respect to the highest-logit token. Specifically, NCI measures the proximity of the hidden representation to the weight vector corresponding to the highest logit, while fDBD treats the representation as residing within the decision region of that token of max logit—a notion that may become less precise under stochastic decoding.

However, we hypothesize that both metrics would “catch up” over the course of a reasoning task and retain strong performance on a sequence level. The geometric uncertainty measures in NCI and fDBD capture the model’s internal uncertainty at each decoding state. When the generated token aligns with this internal uncertainty—low-probability tokens in uncertain states or high-probability tokens in certain states—the measures accurately reflect per-step generation confidence. Misalignment can occur under stochastic decoding, for example, when sampling produces a low-probability token in an otherwise certain state (the converse case is practically infeasible). Such mismatches are typically short-lived: the model tends to transition into genuinely uncertain states in subsequent steps, allowing the uncertainty signal to quickly recover and remain reliable over the remainder of the sequence.

Table 3 validates our hypothesis and shows strong performance of NCI and fDBD under stochastic decoding. We evaluate performance across a range of standard decoding temperatures $temp \in \{0.2, 0.5, 0.8, 1.0\}$. For each temperature setting, we conduct five independent runs using different random seeds and report the mean AUROC along with its standard deviation. For simplicity, we set alternative set size $k = \text{All}$. As observed in the table, NCI and fDBD maintain strong performance across all temperatures and consistently outperform the baseline Perplexity.

We further validate our hypothesis in Appendix E by considering a decoded-token-based framework as an alternative. While fDBD does not readily extend to this setting, NCI adapts naturally by measuring feature proximity to the decoded token’s weight vector. Empirically, this variant achieves performance comparable to the original NCI, con-

firmed that imprecise alignment during stochastic decoding does not degrade sequence-level performance.

5. Experiments

In this section, we evaluate the hallucination detectors NCI and fDBD across additional datasets and model architectures to assess their generalization. We compare our methods against a broader set of baselines, where NCI and fDBD consistently achieve superior performance. We further analyze the sensitivity of the hyperparameter k , which controls the size of the alternative token set in fDBD.

5.1. Main Results

Datasets In addition to the commonsense reasoning task CSQA, we evaluate on two mathematical reasoning benchmarks: GSM8K (Cobbe et al., 2021), which requires free-form numerical answers, and AQuA (Ling et al., 2017), which uses a multiple-choice format (see Appendix B for examples). We evaluate on the CSQA validation set (1,221 questions), the GSM8K test set (1,319 questions), and the AQuA validation split (254 questions). For each dataset, hyperparameter k , which controls the alternative set size, is selected with training subset.

Models We extend the analysis from Llama-3.2-3B-Instruct (Section 4) to Qwen-2.5-7B-Instruct (Qwen et al., 2025) to assess generality across model families. We further demonstrate scalability to larger models by evaluating Qwen-3-32B (Yang et al., 2025) in Appendix A.

Baselines We compare our method against *training-free* hallucination detectors that operate on a *single sample*. In addition to Perplexity (Section 4), we consider common baselines: Predictive Probability, i.e., $\prod_{t=1}^T p(\mathbf{y}_t | \mathbf{x}, \mathbf{y}_{<t})$, Length-Normalized (LN) Predictive Probability, which normalizes predictive probability by output sequence length; Perplexity (Si et al., 2022), which measures model predictive uncertainty; Maximum Softmax Probability (Max P) (Wang et al., 2024); and verbalized method P(True) (Kadavath et al., 2022), which queries model confidence through an auxiliary prompt verbatim, as in the original paper. Following (Wang et al., 2024), we also consider latent-space

Table 4. OOD inspired hallucination detectors NCI and fDBD achieve superior performance across different datasets and model architectures. Our methods, NCI, fDBD with k set to ALL, and fDBD with k selected on the validation set, are highlighted in shading. “Single Sample” indicates whether the method requires multiple samples for hallucination detection. All methods are *training-free*. Performance is reported in AUROC (higher is better). Best performance is shown in **bold**, second best is underlined.

Model	Methods	Single Sample	CSQA	GSM8K	AQuA
Llama-3.2-3B-Instruct	Perplexity	✓	63.23	69.63	72.85
	Predictive Probability	✓	61.63	70.88	69.07
	LN Predictive Probability	✓	61.51	70.68	68.98
	Max P	✓	66.01	73.90	66.02
	P(True)	✓	47.73	51.02	39.38
	CoE-R	✓	47.06	50.12	45.55
	CoE-C	✓	58.82	60.69	62.56
	Lexical Similarity	✗	62.94	73.66	71.48
	SelfCheckGPT NLI	✗	64.18	74.29	66.01
	Semantic Entropy	✗	60.61	64.40	64.71
	NCI	✓	66.07	<u>76.32</u>	74.41
	fDBD	✓	<u>68.15</u>	75.59	<u>75.80</u>
	fDBD (selected k)	✓	69.24	76.36	76.20
Qwen-2.5-7B-Instruct	Perplexity	✓	61.94	71.54	71.66
	Predictive Probability	✓	64.91	73.29	73.37
	LN Predictive Probability	✓	65.19	73.01	74.17
	Max P	✓	49.90	50.00	50.83
	P(True)	✓	68.01	70.31	72.86
	CoE-R	✓	62.75	75.13	72.13
	CoE-C	✓	66.89	75.50	72.04
	Lexical Similarity	✗	60.57	72.02	72.62
	SelfCheckGPT NLI	✗	60.18	76.22	70.90
	Semantic Entropy	✗	59.10	66.83	69.62
	NCI	✓	<u>71.60</u>	75.83	<u>78.19</u>
	fDBD	✓	71.50	<u>77.19</u>	77.08
	fDBD (selected k)	✓	72.47	77.19	78.22

Chain-of-Embeddings (CoE) methods, including CoE-R and CoE-C.

In addition, we compare against baselines that detect hallucinations by measuring consistency across *multiple samples*. Specifically, we compare against Lexical Similarity (Lin et al., 2022; 2023), which measures consistency using ROUGE-L scores; SelfCheckGPT-NLI (Manakul et al., 2023), which measures discrepancies via contradiction scores from natural language inference; and Semantic Entropy (Kuhn et al., 2023b), which clusters similar texts and then computes entropy across clusters. In our experiments, we set the sample size to three and the temperature to 0.5. The rest of the setups follow the original paper.

Performance Along with the baselines, we evaluate our hallucination detectors NCI and fDBD. All experiments are conducted under greedy decoding ($T = 0$), and AUROC is reported, where higher values indicate better performance. We note that AUROC is conventionally reported in the range

[50, 100]; following prior work (Wang et al., 2024), we allow values outside this range when hallucinated samples receive higher confidence scores than non-hallucinated ones, to facilitate comparison across methods.

For fDBD, we consider two variants: the hyperparameter-free version that considers distances to all decision boundaries, and a version that computes distances only to the top k candidate tokens, with $k \in \{1, 10, 100, 1000, 10000, 100000, \text{All}\}$. Based on validation results, we select $k = 1000, 100, 100$ on Llama-3.2-3B-Instruct for CSQA, GSM8K, and AQuA respectively; and $k = 100, \text{All}, 100$ on Qwen-2.5-7B-Instruct for CSQA, GSM8K, and AQuA respectively;

Examining the results, fDBD with a selected subset of size k almost always outperforms the full fDBD variant, confirming the benefit of controlling the size of the alternative token set. In most cases, fDBD also outperforms NCI. This reflects the efficiency–effectiveness trade-off between the two

Table 5. OOD inspired hallucination detector fDBD exhibits low sensitivity to the choice of k , which controls the size of the alternative token set. AUROC is reported. Experiments are conducted on Llama-3.2-3B-Instruct. The selected values of k are 1000 for CSQA and 100 for GSM8K and AQuA.

Dataset	Variation in k						
	-20%	-10%	-5%	Selected (k)	+5%	+10%	+20%
CSQA	69.25	69.24	69.24	69.24	69.23	69.23	69.22
GSM8K	76.41	76.38	76.36	76.36	76.34	76.34	76.32
AQuA	76.34	76.26	76.26	76.20	76.21	76.23	76.15

methods: NCI has per-step complexity $O(d_{\text{model}})$, making it computationally cheaper than fDBD, which has complexity $O(d_{\text{model}}|\mathcal{V}|)$ in its full form and $O(d_{\text{model}}k+|\mathcal{V}|)$ when restricted to a selected subset of size k . Both NCI and fDBD consistently outperform single-sample baselines.

We further observe that multiple-sampling-based methods—Lexical Similarity, SelfCheckGPT NLI, and Semantic Entropy—are less effective on reasoning tasks compared to their original scope of concise question-answering. This supports our intuition that measuring discrepancies across samples is intrinsically challenging in the presence of diverse reasoning chains. In contrast, our methods consistently surpass these approaches in the vast majority of cases, demonstrating strong reliability and computational efficiency across models.

5.2. Hyper-parameter Sensitivity

We next examine the sensitivity of fDBD performance to the choice of k , which controls the size of the alternative token set used for computing the decision boundary distance. To this end, we analyze variations of k around the selected value for each dataset on *Llama-3.2-3B-Instruct*, where the selected k is 1000 for CSQA and 100 for GSM8K and AQuA. Specifically, we evaluate k values obtained by perturbing the selected value by $\pm 5\%$, $\pm 10\%$, and $\pm 20\%$. As shown in Table 5, fDBD performance remains highly stable across all tested variations, with AUROC changing only marginally as k varies. These results validate that fDBD is not overly sensitive to the exact choice of k . Selecting the order of magnitude of k , as done in Section 5.1, is sufficient, making it possible to transfer a value of k selected on the validation set to the test set.

6. Related Work

6.1. Out-of-Distribution Detection

In the context of classifiers, extensive prior work has investigated out-of-distribution (OOD) detection, with the goal of identifying test samples whose classes were unseen during training. In particular, one line of work builds on pre-trained classifiers to design post-hoc OOD detectors.

For example, (Hendrycks et al., 2019; Liang et al., 2018; Liu et al., 2020; Sun et al., 2021; Sun & Li, 2022; Xu et al., 2024) define OOD scores in the output space of a classifier, whereas (Lee et al., 2018; Sun et al., 2022; Liu & Qin, 2024; 2025) measure OOD-ness in the feature space. In this work, we conceptually bridge OOD detection in classifiers to hallucination detection in LLMs. Specifically, we extend prior work on OOD detection, NCI(Liu & Qin, 2025) and fDBD(Liu & Qin, 2024), to the task of hallucination detection and demonstrate its strong performance in this setting.

6.2. Hallucination Detection

Hallucinations pose risks to the safe deployment of LLMs, and much existing research has therefore focused on detecting them at inference time. One line of work detects hallucinations by measuring consistency across multiple samples (Lin et al., 2023; 2022; Manakul et al., 2023; Xiao & Wang, 2021; Kuhn et al., 2023a; Chen et al., 2024; Liu et al., 2025; Hou et al., 2024; Jiang et al., 2023; Gao et al., 2024). While effective, such methods introduce nontrivial inference overhead due to the need for repeated sampling. Another line of work aims to detect hallucinations from a single inference without repetitive sampling (Azaria & Mitchell; Kossen et al., 2024; Li et al., 2023; Liu et al.; Manakul et al., 2023; Marks & Tegmark; Su et al., 2024; Wang et al., 2024). These methods usually require training a separate model, adding additional computational cost and suffering from training-test distribution shift. To address the limitations of both lines of work, we detect hallucinations by analyzing the distance between internal features and the model’s decision boundary, yielding an inference-efficient and distribution-shift-robust approach.

7. Conclusion

This work tackles hallucination detection in LLMs. We adapt existing OOD detectors to account for the structural differences in large language models, yielding training-free, single-sample detectors that effectively identify hallucinations in reasoning tasks. We hope this framework inspires further OOD-based approaches to enhance LLM safety.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Azaria, A. and Mitchell, T. The internal state of an llm knows when it’s lying. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Chen, C., Liu, K., Chen, Z., Gu, Y., Wu, Y., Tao, M., Fu, Z., and Ye, J. INSIDE: LLMs’ internal states retain the power of hallucination detection. In *The Twelfth International Conference on Learning Representations*, 2024. URL <https://openreview.net/forum?id=Zj12nzlQbz>.

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Gao, X., Zhang, J., Mouatadid, L., and Das, K. Spuq: Perturbation-based uncertainty quantification for large language models. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2336–2346, 2024.

Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Vaughan, A., et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Hendrycks, D., Basart, S., Mazeika, M., Mostajabi, M., Steinhardt, J., and Song, D. Scaling out-of-distribution detection for real-world settings. *arXiv preprint arXiv:1911.11132*, 2019.

Hendrycks, D., Basart, S., Mazeika, M., Zou, A., Kwon, J., Mostajabi, M., Steinhardt, J., and Song, D. Scaling out-of-distribution detection for real-world settings. In *International Conference on Machine Learning*, pp. 8759–8773. PMLR, 2022.

Hoare, C. A. Algorithm 65: find. *Communications of the ACM*, 4(7):321–322, 1961.

Hou, B., Liu, Y., Qian, K., Andreas, J., Chang, S., and Zhang, Y. Decomposing uncertainty for large language models through input clarification ensembling. In *ICML*, 2024.

Jiang, M., Ruan, Y., Huang, S., Liao, S., Pitis, S., Grosse, R. B., and Ba, J. Calibrating language models via augmented prompt ensembles. 2023.

Kadavath, S., Conerly, T., Askell, A., Henighan, T., Drain, D., Perez, E., Schiefer, N., Hatfield-Dodds, Z., DasSarma, N., Tran-Johnson, E., et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Kossen, J., Han, J., Razzak, M., Schut, L., Malik, S., and Gal, Y. Semantic entropy probes: Robust and cheap hallucination detection in llms. *arXiv preprint arXiv:2406.15927*, 2024.

Kuhn, L., Gal, Y., and Farquhar, S. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *The Eleventh International Conference on Learning Representations*, 2023a. URL <https://openreview.net/forum?id=VD-AYtP0dve>.

Kuhn, L., Gal, Y., and Farquhar, S. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *The Eleventh International Conference on Learning Representations*, 2023b. URL <https://openreview.net/forum?id=VD-AYtP0dve>.

Lee, K., Lee, K., Lee, H., and Shin, J. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in neural information processing systems*, 31, 2018.

Li, J., Cheng, X., Zhao, W. X., Nie, J.-Y., and Wen, J.-R. Halueval: A large-scale hallucination evaluation benchmark for large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 6449–6464, 2023.

Liang, S., Li, Y., and Srikant, R. Enhancing the reliability of out-of-distribution image detection in neural networks. In *6th International Conference on Learning Representations, ICLR 2018*, 2018.

Lin, Z., Liu, J. Z., and Shang, J. Towards collaborative neural-symbolic graph semantic parsing via uncertainty. *Findings of the Association for Computational Linguistics: ACL 2022*, 2022.

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

Ling, W., Yogatama, D., Dyer, C., and Blunsom, P. Program induction by rationale generation: Learning to solve and explain algebraic word problems. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 158–167, 2017.

- 495 Liu, K., Casper, S., Hadfield-Menell, D., and Andreas, J.
 496 Cognitive dissonance: Why do language model outputs
 497 disagree with internal representations of truthfulness? In
 498 *The 2023 Conference on Empirical Methods in Natural*
 499 *Language Processing*.
- 500 Liu, L. and Qin, Y. Fast decision boundary based out-of-
 501 distribution detector. *arXiv preprint arXiv:2312.11536*,
 502 2023.
- 503 Liu, L. and Qin, Y. Fast decision boundary based out-of-
 504 distribution detector. In *Forty-first International Confer-*
 505 *ence on Machine Learning*, 2024.
- 506 Liu, L. and Qin, Y. Detecting out-of-distribution through the
 507 lens of neural collapse. In *Proceedings of the Computer*
 508 *Vision and Pattern Recognition Conference*, pp. 15424–
 509 15433, 2025.
- 510 Liu, L., Pourreza, R., Panchal, S., Bhattacharyya, A., Qin,
 511 Y., and Memisevic, R. Enhancing hallucination detection
 512 through noise injection. *arXiv preprint arXiv:2502.03799*,
 513 2025.
- 514 Liu, W., Wang, X., Owens, J., and Li, Y. Energy-based out-
 515 of-distribution detection. *Advances in Neural Information*
 516 *Processing Systems*, 33:21464–21475, 2020.
- 517 Liu, X., Lochman, Y., and Zach, C. Gen: Pushing the
 518 limits of softmax-based out-of-distribution detection. In
 519 *Proceedings of the IEEE/CVF Conference on Computer*
 520 *Vision and Pattern Recognition*, pp. 23946–23955, 2023.
- 521 Malinin, A. and Gales, M. Uncertainty estimation in
 522 autoregressive structured prediction. *arXiv preprint*
 523 *arXiv:2002.07650*, 2020.
- 524 Manakul, P., Liusie, A., and Gales, M. SelfCheckGPT:
 525 Zero-resource black-box hallucination detection for gen-
 526 erative large language models. In *Proceedings of the 2023*
 527 *conference on empirical methods in natural language pro-*
 528 *cessing*, pp. 9004–9017, 2023.
- 529 Marks, S. and Tegmark, M. The geometry of truth: Emer-
 530 gent linear structure in large language model represen-
 531 tations of true/false datasets. In *First Conference on*
 532 *Language Modeling*.
- 533 Qwen, :, Yang, A., Yang, B., Zhang, B., Hui, B., Zheng,
 534 B., Yu, B., Li, C., Liu, D., Huang, F., Wei, H., Lin, H.,
 535 Yang, J., Tu, J., Zhang, J., Yang, J., Yang, J., Zhou, J.,
 536 Lin, J., Dang, K., Lu, K., Bao, K., Yang, K., Yu, L.,
 537 Li, M., Xue, M., Zhang, P., Zhu, Q., Men, R., Lin, R.,
 538 Li, T., Tang, T., Xia, T., Ren, X., Ren, X., Fan, Y., Su,
 539 Y., Zhang, Y., Wan, Y., Liu, Y., Cui, Z., Zhang, Z., and
 540 Qiu, Z. Qwen2.5 technical report, 2025. URL <https://arxiv.org/abs/2412.15115>.
- 541 Ren, J., Luo, J., Zhao, Y., Krishna, K., Saleh, M., Lakshmi-
 542 narayanan, B., and Liu, P. J. Out-of-distribution detection
 543 and selective generation for conditional language models.
 544 In *The Eleventh International Conference on Learning*
 545 *Representations*, 2023.
- 546 Si, C., Gan, Z., Yang, Z., Wang, S., Wang, J., Boyd-Graber,
 547 J., and Wang, L. Prompting gpt-3 to be reliable. *arXiv*
 548 *preprint arXiv:2210.09150*, 2022.
- 549 Su, W., Wang, C., Ai, Q., Hu, Y., Wu, Z., Zhou, Y., and
 Liu, Y. Unsupervised real-time hallucination detection
 based on the internal states of large language models. In
Findings of the Association for Computational Linguistics
ACL 2024, pp. 14379–14391, 2024.
- Sun, Y. and Li, Y. Dice: Leveraging sparsification for out-
of-distribution detection. In *European Conference on*
Computer Vision, pp. 691–708. Springer, 2022.
- Sun, Y., Guo, C., and Li, Y. React: Out-of-distribution
detection with rectified activations. *Advances in Neural*
Information Processing Systems, 34:144–157, 2021.
- Sun, Y., Ming, Y., Zhu, X., and Li, Y. Out-of-distribution
detection with deep nearest neighbors. *arXiv preprint*
arXiv:2204.06507, 2022.
- Talmor, A., Herzig, J., Lourie, N., and Berant, J. Common-
senseQA: A question answering challenge targeting com-
monsense knowledge. In *Proceedings of the 2019 Confer-*
ence of the North American Chapter of the Association for
Computational Linguistics: Human Language Technolo-
gies, Volume 1 (Long and Short Papers), pp. 4149–4158,
Minneapolis, Minnesota, June 2019. Association for
Computational Linguistics. doi: 10.18653/v1/N19-1421.
URL <https://aclanthology.org/N19-1421>.
- Wang, H., Li, Z., Feng, L., and Zhang, W. Vim: Out-of-
distribution with virtual-logit matching. In *Proceedings of*
the IEEE/CVF conference on computer vision and pattern
recognition, pp. 4921–4930, 2022.
- Wang, Y., Zhang, P., Yang, B., Wong, D. F., and Wang, R.
Latent space chain-of-embedding enables output-free llm
self-evaluation. *arXiv preprint arXiv:2410.13640*, 2024.
- Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi,
E., Le, Q. V., Zhou, D., et al. Chain-of-thought prompting
elicits reasoning in large language models. *Advances in*
neural information processing systems, 35:24824–24837,
2022.
- Xiao, Y. and Wang, W. Y. On hallucination and predictive
uncertainty in conditional language generation. In *Pro-*
ceedings of the 16th Conference of the European Chapter
of the Association for Computational Linguistics: Main
Volume, 2021.

550 Xu, C., Yu, F., Xu, Z., Inkawhich, N., and Chen, X. Out-
551 of-distribution detection via deep multi-comprehension
552 ensemble. *arXiv preprint arXiv:2403.16260*, 2024.

553
554 Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng,
555 B., Yu, B., Gao, C., Huang, C., Lv, C., Zheng, C., Liu,
556 D., Zhou, F., Huang, F., Hu, F., Ge, H., Wei, H., Lin,
557 H., Tang, J., Yang, J., Tu, J., Zhang, J., Yang, J., Yang,
558 J., Zhou, J., Zhou, J., Lin, J., Dang, K., Bao, K., Yang,
559 K., Yu, L., Deng, L., Li, M., Xue, M., Li, M., Zhang,
560 P., Wang, P., Zhu, Q., Men, R., Gao, R., Liu, S., Luo,
561 S., Li, T., Tang, T., Yin, W., Ren, X., Wang, X., Zhang,
562 X., Ren, X., Fan, Y., Su, Y., Zhang, Y., Zhang, Y., Wan,
563 Y., Liu, Y., Wang, Z., Cui, Z., Zhang, Z., Zhou, Z., and
564 Qiu, Z. Qwen3 technical report, 2025. URL <https://arxiv.org/abs/2505.09388>.
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604

A. Scalability to Larger Language Model

To evaluate the scalability of our approach to larger language models, we conduct experiments on Qwen-3-32B (Yang et al., 2025). For computational concerns, we focus on three baselines: Perplexity, Predictive Probability, and Length-Normalized (LN) Predictive Probability. For fDBD, we did not do hyperparameter sweeping of k and report with all k . We focus on three computationally efficient As shown in Table 6, NCI achieves strong overall performance, while fDBD consistently outperforms these baselines across CSQA, GSM8K, and AQuA. These results demonstrate that methods adapted from OOD detection scale favorably to large language models.

Table 6. **OOD-inspired hallucination detectors scale to larger language models.** Results on Qwen-3-32B. We report AUROC (higher is better), with the best performance highlighted in **bold**. NCI achieves strong overall performance, while fDBD consistently outperforms the baselines across datasets.

Methods	CSQA	GSM8K	AQuA
Perplexity	58.44	76.86	53.89
Predictive Probability	59.30	74.99	57.97
LN Predictive Probability	59.35	74.38	58.21
NCI	64.54	75.75	60.28
fDBD	65.68	80.60	65.37

B. Implementation details

B.1. Examples of Questions and Model Responses for each Dataset

B.1.1. CSQA EXAMPLE

Question: A revolving door is convenient for two direction travel, but it also serves as a security measure at a what?

- (a) bank
- (b) library
- (c) department store
- (d) mall
- (e) new york

Model response: The answer should be a place where security is a priority. Of the above choices, a bank is the most likely to use a revolving door as a security measure. So the answer is (a).

B.1.2. GSM8K EXAMPLE

Question: Janet’s ducks lay 16 eggs per day. She eats three for breakfast every morning and bakes muffins for her friends every day with four. She sells the remainder at the farmers’ market daily for \$2 per fresh duck egg. How much in dollars does she make every day at the farmers’ market?

Model response: Janet’s ducks lay 16 eggs per day. She uses 3 for breakfast and 4 for baking, which totals $3 + 4 = 7$ eggs used daily. The remaining eggs are $16 - 7 = 9$. She sells these for \$2 each, so $9 \times 2 = 18$. The answer is 18.

B.1.3. AQUA EXAMPLE

Question: A password needs to contain 2 letters and 3 numbers. How many different passwords are possible if repetition of letters and numbers is allowed??

- (a) 676000
- (b) 676
- (c) 100

(d) 6760

(e) 25

Model response: There are 26 letters in the alphabet and 10 digits (0-9). Since repetition is allowed, there are $26 * 26 = 676$ ways to choose the letters and $10 * 10 * 10 = 1000$ ways to choose the numbers. The total number of different passwords is $676 * 1000 = 676000$. The answer is (a).

B.2. Chain-of-Thought Few-shot Prompting

Following (Wei et al., 2022), we use chain-of-thought few-shot prompting to elicit reasoning in the model’s responses. For all datasets, we adopt the canonical examples from (Wei et al., 2022). These prompts guide the model to produce a formatted answer after completing the reasoning process. For answer extraction, we extract the letter following “*So the answer is*” for CSQA, the number following “*The answer is*” for GSM8K, and the answer letter following “*The answer is*” for AQuA. An answer is considered correct if the extracted answer matches the ground truth.

C. Proof of Theorem 3.3

Proof. Let $\hat{c} = \arg \max_{v \in \mathcal{V}} \mathbf{w}_v^\top \mathbf{z} + b_v$ denote the model’s most confident internal prediction for embedding \mathbf{z} . For any alternative token $c \in \mathcal{V}$ with $c \neq \hat{c}$, recall that the decision region of token c is

$$\mathcal{R}_c = \{ \mathbf{z} \in \mathbb{R}^{d_{\text{model}}} : \mathbf{w}_c^\top \mathbf{z} + b_c > \mathbf{w}_v^\top \mathbf{z} + b_v, \forall v \neq c \}.$$

We further define a relaxed region

$$\mathcal{R}'_c = \{ \mathbf{z} \in \mathbb{R}^{d_{\text{model}}} : \mathbf{w}_c^\top \mathbf{z} + b_c > \mathbf{w}_{\hat{c}}^\top \mathbf{z} + b_{\hat{c}} \},$$

which only enforces that token c surpasses the current most confident token \hat{c} . By construction, $\mathcal{R}_c \subseteq \mathcal{R}'_c$. Therefore,

$$D_f(\mathbf{z}, c) = \inf_{\mathbf{z}' \in \mathcal{R}_c} \|\mathbf{z}' - \mathbf{z}\|_2 \geq \inf_{\mathbf{z}' \in \mathcal{R}'_c} \|\mathbf{z}' - \mathbf{z}\|_2. \quad (8)$$

Geometrically, the quantity $\inf_{\mathbf{z}' \in \mathcal{R}'_c} \|\mathbf{z}' - \mathbf{z}\|_2$ corresponds to the Euclidean distance from \mathbf{z} to the hyperplane

$$(\mathbf{w}_{\hat{c}} - \mathbf{w}_c)^\top \mathbf{z} + (b_{\hat{c}} - b_c) = 0. \quad (9)$$

Thus,

$$\inf_{\mathbf{z}' \in \mathcal{R}'_c} \|\mathbf{z}' - \mathbf{z}\|_2 = \frac{|(\mathbf{w}_{\hat{c}} - \mathbf{w}_c)^\top \mathbf{z} + (b_{\hat{c}} - b_c)|}{\|\mathbf{w}_{\hat{c}} - \mathbf{w}_c\|_2}. \quad (10)$$

Combining (8) and (10) yields the lower bound in Eqn. (3).

Finally, let

$$c_2 = \arg \min_{c \in \mathcal{V}, c \neq \hat{c}} \inf_{\mathbf{z}' \in \mathcal{R}'_c} \|\mathbf{z}' - \mathbf{z}\|_2 \quad (11)$$

denote the token whose associated hyperplane is closest to \mathbf{z} , and let \mathbf{p} be the projection of \mathbf{z} onto this hyperplane. Since all other hyperplanes are farther from \mathbf{z} than \mathbf{p} , the points \mathbf{z} and \mathbf{p} lie on the same side of every such hyperplane. Consequently, \mathbf{p} lies in the closure of \mathcal{R}_{c_2} , implying

$$\|\mathbf{p} - \mathbf{z}\|_2 \geq \inf_{\mathbf{z}' \in \mathcal{R}_{c_2}} \|\mathbf{z}' - \mathbf{z}\|_2. \quad (12)$$

Together with the definition of c_2 , this shows that the lower bound in Eqn. (3) is tight. \square

D. Derivative of Decision-Neutral Point

Formally, for a language head, a decision-neutral point \mathbf{z}_* is an embedding at which all tokens receive identical logits, i.e.,

$$\mathbf{w}_v^\top \mathbf{z}_* + b_v = \mathbf{w}_{v'}^\top \mathbf{z}_* + b_{v'}, \forall v, v' \in \mathcal{V}$$

However, the resulting system of linear equations may be overdetermined and thus fail to admit a feasible solution. We therefore relax this notion and define the decision-neutral closest point.

Definition D.1 (Decision-Neutral Closest Point). Given a classification head f_{head} , a *decision-neutral closest point* is the embedding that minimizes the variance of logits across the entire vocabulary:

$$\hat{z}_* = \arg \min_z \sum_{v \in \mathcal{V}} (\mathbf{w}_v^\top z + b_v - \bar{l})^2, \quad (13)$$

where $\bar{l} = \frac{1}{|\mathcal{V}|} \sum_{v \in \mathcal{V}} (\mathbf{w}_v^\top z + b_v)$ represents the mean logit value across the vocabulary.

To derive analytical solution in Equation 4, we first convert the definition above to matrix form:

$$\hat{z}_* = \arg \min_z \|P(Wz + \mathbf{b})\|^2 \quad (14)$$

where $P = I - \frac{1}{n} \mathbf{1}\mathbf{1}^\top$ is the symmetric and idempotent centering matrix. Define the function of z as

$$f(z) = \|P(Wz + \mathbf{b})\|^2 \quad (15)$$

To find \hat{z}_* , we now compute the gradient with respect to z :

$$\nabla_z f(z) = (PW)^\top \cdot 2[P(Wz + \mathbf{b})] \quad (16)$$

$$= 2W^\top P^\top P(Wz + \mathbf{b}) \quad (17)$$

Setting the gradient to zero and we have:

$$\nabla_z f(z) = 2W^\top P(Wz + \mathbf{b}) \quad (18)$$

The analytical solution is thus:

$$\hat{z}_* = -(W^\top PW)^\dagger W^\top P\mathbf{b}, \quad (19)$$

where $(\cdot)^\dagger$ denotes the Moore–Penrose pseudo-inverse.

E. Decoded-token based Variant of NCI

In addition to computing the NCI score with respect to the highest-logit token, we evaluate a variant where feature proximity is computed against the weight vector of the actually decoded token at each step. Under greedy decoding, this setup is the same as the adopted NCI in the main paper. Under stochastic decoding, this setup aligns the metric more directly with the stochastic output of the model. We evaluate performance across a range of standard decoding temperatures $temp \in \{0.2, 0.5, 0.8, 1.0\}$. For each temperature setting, we conduct five independent runs using different random seeds and report the mean AUROC along with its standard deviation. As shown in Table 7, the performance of this variant is similar to the max-logit version of NCI.

Table 7. Two NCI variants, max logit token based and decoded token based, exhibit similar performance under stochastic decoding. Results on CSQA using Llama-3.2-3B-Instruct. AUROC is reported (higher is better). For each temperature, mean \pm standard deviation is computed over five random seeds.

Method	temp = 0.2	temp = 0.5	temp = 0.8	temp = 1.0
NCI (max-logit token)	67.07 \pm 0.53	66.04 \pm 1.59	67.53 \pm 0.88	67.93 \pm 1.65
NCI (decoded token)	67.10 \pm 0.52	66.04 \pm 1.63	67.44 \pm 0.86	67.70 \pm 1.53