DETECTING LLM HALLUCINATIONS VIA NONLINEAR MANIFOLD SEPARATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models (LLMs) demonstrate exceptional performance across various tasks but are prone to hallucinations, raising concerns about the reliability of their outputs. When dealing with hallucination issues in unlabeled data within wild settings, existing approaches have suggested using latent feature space information for detecting hallucinations. However, these studies have not thoroughly analyzed the distribution of samples within the latent feature space and typically rely on linear methods. To better understand data distributions, we introduce Hallucination Attention Regions (HARs) and True Attention Regions (TARs) to describe the latent-space distributions of hallucinated and truthful samples. Our empirical analysis confirms that HARs and TARs are nonlinearly separable. Based on the hypothesis that high-dimensional sample distributions can be embedded into a low-dimensional manifold, we propose the HDME framework, which enables the automatic detection of hallucination samples in unlabeled data. The HDME framework involves three steps: (1) embedding high-dimensional samples into low-dimensional manifolds, (2) clustering the data to generate pseudo-labels, and (3) training a hallucination detector using these labels. Experimental results demonstrate that our method achieves superior performance in hallucination detection across diverse datasets. We will release the code upon acceptance.

1 INTRODUCTION

The emergence of large language models (LLMs) Zhao et al. (2023) has transformed the field of natural language processing (NLP), driving major progress in text generation, comprehension, and reasoning tasks. Trained on vast datasets, LLMs can generate coherent and contextually relevant text. However, a key challenge is the hallucination phenomenon Ji et al. (2023). LLMs sometimes produce plausible but factually incorrect information, which can appear as made-up facts, misattributed quotes, or completely fabricated stories. This undermines the reliability of LLMs, especially in applications where accuracy is crucial, like legal documentation, medical advice, and educational content. Thus, hallucination detection Luo et al. (2024), which determines the truthfulness of LLM-generated content, is of great importance.

The previous method, Haloscope Du et al. (2024), aims at the issue of unlabeled data in real-world scenarios and proposes a more challenging setting, which is the hallucination detection problem under wild data. The core idea of Haloscope is to make use of the language model's latent representations that can capture information related to truthfulness. Specifically, it identifies a subspace for hallucinated statements in the activation space and considers a point potentially hallucinated if its representation aligns strongly with this subspace. Factorization of LLM embeddings is used, with top singular vectors forming a latent subspace for membership estimation. The estimation score, measuring the projected embedding norm on these vectors, varies for different data types, has a simple math interpretation, and is easy to implement. However, Haloscope does not thoroughly analyze the distribution of samples within the latent feature space and relies on linear methods.

To better understand the data distribution, inspired by Crabbé & van der Schaar (2022) who proposed Concept Activation Regions (CARs) using kernel tricks to explain the distribution of different concepts in the latent space, we propose Hallucination Attention Regions (HARs) and True Attention Regions (TARs) for hallucinated and truthful samples respectively. This enables us to comprehend the distribution of hallucinated and truthful information generated by LLMs in their latent feature space.

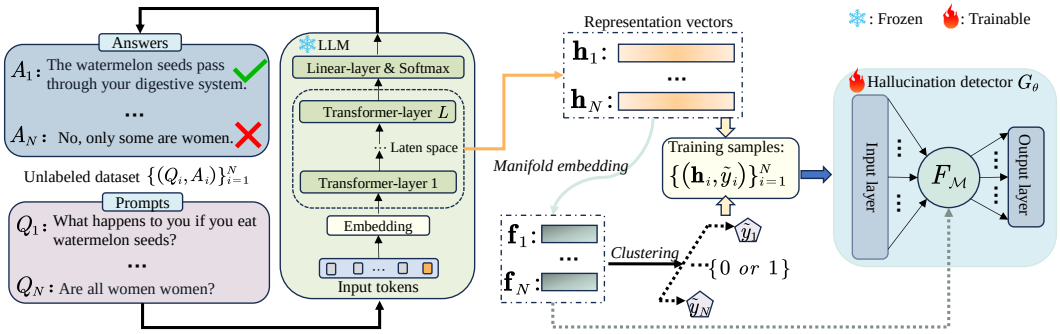


Figure 1: Overview of the proposed HDME framework. The process begins by generating an unlabeled dataset through the deployment of an LLM in real-world scenarios. Representation vectors are then extracted from the model's outputs. HDME employs nonlinear manifold embedding to project these vectors into a low-dimensional subspace, where clustering is performed to generate pseudo-labels. Finally, a manifold submodule $F_\mathcal{M}$ is incorporated into the hallucination detector G_θ, as detailed in Section 5.3.

We assume that HARs and TARs are non-linearly separable in the LLM latent space, which is in line with the concept smoothness assumption of CARs. Analysis of the TruthfulQA Lin et al. (2022a) dataset in the latent space of Qwen-2.5-7b Qwen et al. (2025) provides support for this assumption.

In this paper, we propose a novel LLM hallucination detection method named HDME. High-dimensional data frequently contain irrelevant or noisy features, which can significantly impede the accuracy and efficiency of data analysis. To address this issue, the application of nonlinear manifold fitting methods proves to be an effective approach for eliminating redundancy Meilă & Zhang (2024). Given the assumption that high-dimensional data can be embedded into a low-dimensional manifold Lin & Zha (2008b), our proposed HDME method leverages the structural and geometric information of samples in the latent space to improve the detection of hallucination samples in unlabeled data. Specifically, HDME first embeds high-dimensional representation samples into low-dimensional manifolds using manifold fitting, then clusters the embedded data to generate pseudo-labels, and finally uses these labels to train a hallucination detector.

Extensive experimental results across various series and scales of large language models demonstrate that our method, HDME, surpasses HaloScope and several other advanced methods in hallucination detection performance across multiple datasets. Furthermore, HDME exhibits more stable results than HaloScope in multiple independent repeated experiments. For instance, using Llama2-7B-chat, our method achieves a cumulative improvement of 20.27% in mean AUROC across four different datasets compared to HaloScope, along with a cumulative reduction of 14.28% in standard deviation. Additionally, we conducted ablation studies on the critical modules and hyperparameters of HDME to thoroughly explore the function of each module, aiding in future enhancements.

We summarize our contributions as follows:

- To better understand data distributions in the latent space, we introduce Hallucination Attention Regions (HARs) and True Attention Regions (TARs) to analyze the distribution of hallucinated and truthful samples in the latent space of LLMs. Our empirical analysis shows that HARs and TARs are non-linearly separable. (See Section 4)
- We propose HDME, a novel framework that uses manifold embedding and clustering to automatically detect hallucinated samples in unlabeled data, enhancing the performance and robustness of hallucination detection in the latent space.(See Section 5)
- Experimental results demonstrate that our method achieves superior performance in hallucination detection across diverse datasets. Besides, we conduct in-depth ablation studies on key modules and hyperparameters in HDME. (See Section 6)

2 Related Work

Hallucination Detection. Detecting hallucinations in Large Language Models (LLMs) is critical for ensuring content reliability. Research in this area has explored diverse approaches. Some methods focus on self-consistency without external data; Manakul et al. (2023) introduced SelfCheckGPT,

which assesses the internal consistency of an LLM's own outputs. Other work analyzes the internal states of LLMs. For example, Chen et al. (2024a) proposed the INSIDE framework, using the EigenScore metric to assess self-consistency from internal representations. Similarly, Su et al. (2024) presented MIND, an unsupervised framework that leverages internal states for real-time detection and reportedly outperforms prior methods on the HELM benchmark. Supervised approaches have also been developed, such as the RelD discriminator by Chen et al. (2023), trained on a bilingual QA dataset. Efforts to advance the field also include creating new datasets and benchmarks. Liu et al. (2021) proposed HADES, a token-level dataset for fine-grained, reference-free detection. Chen et al. (2024b) introduced the FACTCHD benchmark for fact-conflicting hallucinations and TRUTH-TRIANGULATOR, a method that combines tool-use with LoRA-tuning. Kang et al. (2024) evaluated metrics in multilingual settings, finding NLI-based approaches more effective than lexical ones, though challenges remain for low-resource languages. Finally, Park et al. (2025) propose Truthfulness Separator Vector (TSV) to reshape LLMs' latent space during inference for separating truthful and hallucinated outputs, via two-stage training with small labeled data and OT-based pseudo-labeling.

Manifold fitting. Manifold learning provides foundational techniques for representing high-dimensional data in lower-dimensional spaces. Early work by Belkin & Niyogi (2003) used the graph Laplacian to construct locality-preserving embeddings. Subsequently, Fefferman et al. (2018; 2023) developed algorithms with theoretical guarantees to fit manifolds from noisy data, ensuring a small Hausdorff distance to the ground-truth manifold. These foundational methods have been extended to diverse applications and advanced architectures. For example, recent work has applied manifold fitting to denoise single-cell RNA sequencing data for improved clustering (Yao et al., 2024a), integrated it with generative adversarial networks to map latent spaces to data manifolds (Yao et al., 2024b), and extended it to non-Euclidean spaces with sample-efficient guarantees for noisy observations (Yao et al., 2023).

3 Problem Setting

Following Haloscope Du et al. (2024), this section introduces the generation of LLMs deployed in-the-wild scenarios and provides a formal definition of latent space hallucination detection.

Large language models (LLMs), such as GPT Achiam et al. (2023), when deployed in real-world applications, can generate substantial amounts of text based on user prompts. Considering an L-layer, H-head LLM with a causal decoder structure, where the model dimension is D. We denote the input prompt as $\mathbf{x}_{prompt} = \{x_1, \ldots, x_c\} \subseteq \mathcal{V}$, where $\mathcal{V}$ represents the vocabulary comprising $n_v = |\mathcal{V}|$ tokens. Subsequently, the model generates an output sequence $\mathbf{x}_{output} = \{x_{c+1}, \ldots, x_n\}$ in an autoregressive fashion, we denote $\mathbf{X} = (\mathbf{x}_{prompt}, \mathbf{x}_{output}) \in \mathcal{X}$ as one sample. For each output token x_j, $j \in \{c+1, \ldots, n\}$, let $\mathbf{x}_j = \{x_1, \ldots, x_{j-1}\}$ denote the prefix of x_j.

Definition 3.1 (Generation of LLM)**.** The generation process for the output token x_j, given the prefix $\mathbf{x}_j$, can be described by $x_j = \underset{\mathbf{v}_{j'} \in \mathcal{V}}{\operatorname{argmax}} \mathbf{P}(\mathbf{v}_{j'}|\mathbf{x}_j)$. Here, $\mathbf{v}_{j'}$ denotes the j'-th token in the vocabulary $\mathcal{V}$, and the probability $\mathbf{P}$ is computed as follows:

$$\mathbf{P}(\mathbf{v}_{j'}|\mathbf{x}_j) = \operatorname{softmax}\left(h^L(x_{j-1})\mathbf{W}_o + \mathbf{b}_o\right)_{j'}, \tag{1}$$

where $h^L(x_{j-1}) \in \mathbb{R}^D$ denotes the representation vector of the query token x_{j-1} at the L-th layer of LLM. $\mathbf{W}_o$ and $\mathbf{b}_o$ are the weight and bias parameters of the token prediction head, respectively. The subscript j' of $\operatorname{softmax}(\cdot)$ identifies the j'-th element of the resulting probability vector.

Definition 3.2 (Hallucination of LLM)**.** When a given prompt $\mathbf{x}_{prompt}$ is provided as input to an LLM, it generates two distinct outputs: $\mathbf{x}_{\text{true}}$, which is consistent with the prompt and free from hallucinations, and $\mathbf{x}_{\text{hal}}$, which contradicts or conflicts with the prompt. We define $\mathbf{X}_{\text{true}} = (\mathbf{x}_{prompt}, \mathbf{x}_{\text{true}}) \in \mathcal{X}$ and $\mathbf{X}_{\text{hal}} = (\mathbf{x}_{prompt}, \mathbf{x}_{\text{hal}}) \in \mathcal{X}$ as two sample pairs. Assume the existence of distributions $\mathcal{D}_{\text{true}}$ and $\mathcal{D}_{\text{hal}}$ over the sample space $\mathcal{X}$ such that $\mathbf{X}_{\text{true}} \sim \mathcal{D}_{\text{true}}$ and $\mathbf{X}_{\text{hal}} \sim \mathcal{D}_{\text{hal}}$.

In a wild scenario, the outputs of LLM are determined solely by their model parameters. These outputs typically contain a mixture of factual information and potentially fabricated content, often referred to as hallucinations. In this study, we adopt the Huber contamination model Huber (1992), as employed by Du et al. (2024), to characterize the unlabeled generations of LLM as follows:

Definition 3.3 (Unlabeled sample distribution)**.** Each unlabeled generative sample $\mathbf{X} \in \mathcal{X}$ from an LLM follows a distribution denoted as $\mathcal{D}_{\text{unlabeled}}$. There exists a constant $\mu \in (0, 1]$ such that

$$\mathcal{D}_{\text{unlabeled}} = (1 - \mu)\mathcal{D}_{\text{true}} + \mu\mathcal{D}_{\text{hal}}. \tag{2}$$

It is important to observe that the value of μ is typically small, as most of the generated information is accurate. Specifically, $\mu = 0$ indicates an idealistic situation that the model does not produce any false information.

For each sample $\mathbf{X} \in \mathcal{X}$, the representation matrix at the l-th layer of a LLM can be described as

$$\mathbf{H}^l = \begin{cases} TF^l(\mathbf{H}^{l-1}), & l \geq 1 \\ \text{Emb}(\mathbf{X}), & l = 0 \end{cases} \tag{3}$$

Here, $\text{Emb}(\mathbf{X})$ denotes the embedding matrix of $\mathbf{X}$, and $TF^l(\cdot)$ is the l-th layer transformer block Vaswani (2017) (see Section A.1 for details). In a decoder-only Transformer model, the representation vector of the last token encapsulates information from all preceding tokens. Therefore, we define the representation vector of a sample as follows:

Definition 3.4 (Representation vector)**.** For a given sample $\mathbf{X} \in \mathcal{X}$ containing n tokens, we define the function $h^l : \mathcal{X} \to \mathcal{H} \subseteq \mathbb{R}^D$, which yields the representation vector of the sample at the l-th layer, expressed as $h^l(\mathbf{X}) = \mathbf{h}_n^l$, where $\mathbf{h}_n^l$ represents the n-th row of the embedding matrix $\mathbf{H}^l$, as defined in Equation (3). Note that this vector is the representation of the final token in the given sample.

Building on the previous definitions, hallucination detection based on the representation space (i.e., latent space) can be defined as follows:

Definition 3.5 (Hallucination detection in representation space)**.** Consider an L-layer LLM. Given any prompt $\mathbf{x}_{prompt}$ and its corresponding output $\mathbf{x}_{output}$, which together as a sample $\mathbf{X} = (\mathbf{x}_{prompt}, \mathbf{x}_{output}) \in \mathcal{X}$, we can extract a representation vector $\mathbf{h} = h^l(\mathbf{X}) \in \mathcal{H}$ at the l-th layer of the LLM, where $1 \leq l \leq L$. The aim of hallucination detection in the representation space is to devise a binary classifier $G_\theta : \mathcal{H} \to \{0, 1\}$ such that

$$G_\theta(\mathbf{h}) = \begin{cases} 0, & if \;\; \mathbf{X} \sim \mathcal{D}_{\text{true}} \\ 1, & otherwise \end{cases} \tag{4}$$

4 Understand Distribution of Hallucinations based on Hallucination Attention Regions

To enhance the understanding of hallucination detection in the latent space, we introduce Hallucination Attention Regions (HARs) and examine the distribution of hallucinated and truthful samples. We start by defining the generation of the unlabeled empirical dataset:

Definition 4.1 (Empirical dataset)**.** Consider a set of prompts $\{\mathbf{x}_{prompt_1}, \cdots, \mathbf{x}_{prompt_N}\}$. These prompts are inputted into a locally deployed, offline LLM, resulting in a set of corresponding outputs $\{\mathbf{x}_{output_1}, \cdots, \mathbf{x}_{output_N}\}$. Each pairing of a prompt and its output, represented as $\mathbf{X}_i = (\mathbf{x}_{prompt_i}, \mathbf{x}_{output_i})$, serves as an empirical sample drawn from a distribution $\mathcal{D}_{\text{unlabeled}}$. Consequently, the collection of all empirical samples is denoted as

$$\mathcal{N} := \{\mathbf{X}_i | \mathbf{X}_i \sim \mathcal{D}_{\text{unlabeled}}; i \in [N]\} \subseteq \mathcal{X}. \tag{5}$$

Within this collection, we define the subset of hallucination samples as $\mathcal{N}^h := \{\mathbf{X}_i | \mathbf{X}_i \sim \mathcal{D}_{\text{true}}; i \in [N]\}$, while the subset of clean samples is defined as $\mathcal{N}^c := \{\mathbf{X}_i | \mathbf{X}_i \sim \mathcal{D}_{\text{hal}}; i \in [N]\}$. Utilizing the function h^l defined as Theorem 3.4, we can get the representation vector set $\mathcal{S}$ of set $\mathcal{N}$, which can be denoted as

$$\mathcal{S} := \{\mathbf{h}_i = h^l(\mathbf{X}_i) | \mathbf{X}_i \in \mathcal{N}\} \subseteq \mathcal{H}. \tag{6}$$

Similarly, we can define set $\mathcal{S}^h := \{\mathbf{h}_i = h^l(\mathbf{X}_i) | \mathbf{X}_i \in \mathcal{N}^h\}$ and set $\mathcal{S}^c := \{\mathbf{h}_i = h^l(\mathbf{X}_i) | \mathbf{X}_i \in \mathcal{N}^c\}$ as the representation vector sets of sets $\mathcal{N}^h$ and $\mathcal{N}^c$ respectively.

According to Hofmann et al. (2008), a kernel function $\kappa : \mathcal{H} \times \mathcal{H} \to \mathbb{R}^+$ can be utilized to measure the proximity between $\mathbf{h}_i$ and $\mathbf{h}_j$ within the space $\mathcal{H}$. In this work, we choose the Gaussian kernel function, defined as

$$\kappa(\mathbf{h}_i, \mathbf{h}_j) = \exp\left(-\frac{\| \mathbf{h}_i - \mathbf{h}_j \|_2^2}{2\sigma^2}\right), \tag{7}$$

where σ^2 represents the variance, which determines the connectivity length scale Von Luxburg (2007). Then, we can define the hallucination density as follows:

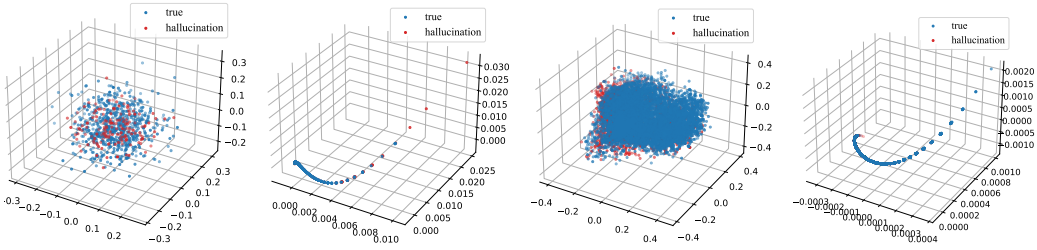
Definition 4.2 (Hallucination density)**.** The hallucination density for each element $\mathbf{h} \in \mathcal{H}$ can be defined as a function $\rho^h : \mathcal{H} \to \mathbb{R}$ such that $\rho^h(\mathbf{h}) := \rho^{\mathcal{S}^h}(\mathbf{h}) - \rho^{\mathcal{S}^c}(\mathbf{h})$, $\rho^{\mathcal{S}^h}(\mathbf{h}) = \frac{1}{N^h}\sum_{\mathbf{h}_i \in \mathcal{S}^h} \kappa(\mathbf{h}, \mathbf{h}_i)$, $\rho^{\mathcal{S}^c}(\mathbf{h}) = \frac{1}{N^c}\sum_{\mathbf{h}_i \in \mathcal{S}^c} \kappa(\mathbf{h}, \mathbf{h}_i)$, where $N^h = |\mathcal{S}^h|$ and $N^c = |\mathcal{S}^c|$. The density of $\mathcal{S}^h$ around the point $\mathbf{h}$ is higher when $\rho^h(\mathbf{h}) > 0$, indicating that $\mathbf{h}$ is more likely to belong to $\mathcal{S}^h$. Conversely, if $\rho^h(\mathbf{h}) < 0$, it suggests that $\mathbf{h}$ is more likely to belong to $\mathcal{S}^c$. It is noteworthy that $\rho^h(\mathbf{h}) \approx 0$ when $\mathbf{h}$ resides at the intersection of $\mathcal{S}^h$ and $\mathcal{S}^c$.

Drawing inspiration from Crabbé & van der Schaar (2022), we apply the density function ρ^h to characterize the distribution of hallucination samples within the LLM representation space as follows:

Definition 4.3 (Hallucination attention region)**.** Consider $\mathcal{H}$ as the representation space of a LLM at the l-th layer. The hallucination attention region (HAR) of the LLM within $\mathcal{H}$ is defined as a subspace $\mathcal{H}^h \subseteq \mathcal{H}$, which satisfies the condition $\mathbb{E}_{\mathbf{h}\in\mathcal{H}^h}[\rho^h(\mathbf{h})] > 0$. The complementary subspace termed the true attention region (TAR), is denoted as $\mathcal{H}^c \subseteq \mathcal{H}$, and satisfies the condition $\mathbb{E}_{\mathbf{h}\in\mathcal{H}^c}[\rho^h(\mathbf{h})] < 0$.

Visualization analysis of hallucinated and truthful samples in the latent space of an open-source LLM (see Figures 2(a) and 2(c)) reveals that their distribution is complex and non-linearly separable. Thus, we make the following assumption, which aligns with the concept smoothness assumption of CARs in the DNN latent space Crabbé & van der Schaar (2022):

Assumption 4.4. The HAR $\mathcal{H}^h$ and the TAR $\mathcal{H}^c$ are nonlinearly separable within the latent space $\mathcal{H}$. In other words, the datasets $\mathcal{S}^h$ and $\mathcal{S}^c$ are nonlinearly separable within $\mathcal{H}$.



(a) Original space (T) (b) Manifold space (T) (c) Original space (Tr) (d) Manifold space (Tr)

Figure 2: The distribution of TruthfulQA (T) Lin et al. (2022a), TriviaQA (Tr) Joshi et al. (2017) samples in the latent feature space of Qwen-2.5-7b. (a) In the original space with dimension $D = 4096$. (b) After embedding onto a manifold with a dimension $d = 3$. Here, we select layer $l = 16$ and use the manifold embedding method referenced in Section 5.1.

5 METHODOLOGY

The complete algorithmic workflow is illustrated in Figure 1, and we now proceed to detail its specific components. We aim to identify hallucination samples within the unlabeled dataset $\mathcal{N}$ using its representation vector set $\mathcal{S}$ in the latent space. The high dimensionality of the LLM latent space likely includes substantial irrelevant or noisy information, complicating the identification of hallucination samples. Consequently, we propose the following assumptions about the latent space data:

Assumption 5.1. The set of representation vectors $\mathcal{S} \subseteq \mathcal{H}$ can be effectively projected onto a low-dimensional manifold $\mathcal{M}$ with dimension d, where $d \ll D$.

The aforementioned assumption is commonly used in the analysis of high-dimensional nonlinear data (Lin & Zha, 2008a; He et al., 2014; Meilă & Zhang, 2024). Additionally, the visualization of the manifold embedding of latent space samples (see Figures 2(b) and 2(d)) further supports this assumption. Based on Theorem 4.4 and Theorem 5.1, we propose a framework, HDME, for detecting hallucinations in the latent space of LLMs. This framework involves three main steps, detailed in Sections 5.1 to 5.3. The whole HDME framework refers to Algorithm 1.

5.1 MANIFOLD EMBEDDING

We first follow Yao et al. (2024a) to process the representation vectors in the original space. The primary objective of this step is to enhance the spatial distribution of the data and capture class-consistent neighborhoods of hallucinated samples.

Manifold projection. Given a set of unlabeled data points $\mathcal{S} = \{\mathbf{h}_i \in \mathbb{R}^D\}_{i=1}^N$, the manifold projection data of $\mathcal{S}$ can be defined as $\mathcal{Z} := \{\mathbf{z}_i = E_{\mathcal{Z}}(\mathbf{h}_i) | \mathbf{h}_i \in \mathcal{S}\}$, where $E_{\mathcal{Z}}$ is a projection function based on manifold fitting:

$$E_{\mathcal{Z}}(\mathbf{h}_i) = \arg\max_{\mathbf{h}_t} \varrho(\mathbf{h}_t), \tag{8}$$

Here, $\mathbf{h}_t$ is a point within the cylinder centered at $\mathbf{h}_i$, $\varrho(\mathbf{h}_t)$ represents the density value of $\mathbf{h}_t$ (see Section A.2). Denote $\bar{\mathbf{z}} = \frac{1}{N}\sum_{i=1}^N \mathbf{z}_i$ as the average vector of $\mathcal{Z}$. The centralized set is then defined as $\hat{\mathcal{Z}} = \{\hat{\mathbf{z}}_i = \mathbf{z}_i - \bar{\mathbf{z}} \mid \mathbf{z}_i \in \mathcal{Z}\}$.

Spectral embedding. To extract the low-dimensional manifold of the dataset, we use the classic spectral method Shi & Malik (2000); Belkin & Niyogi (2003) to project high-dimensional data into a lower-dimensional space. For the dataset $\hat{\mathcal{Z}} = \{\hat{\mathbf{z}}_i \in \mathbb{R}^D\}_{i=1}^N$, we first compute the adjacency matrix $\mathbf{W} \in \mathbb{R}^{N\times N}$ by evaluating the similarity between samples: $\mathbf{W}_{ij} = \kappa(\hat{\mathbf{z}}_i, \hat{\mathbf{z}}_j)$, where κ is the Gaussian kernel function defined in Equation (7). Let $\mathbf{D}$ be the degree matrix, defined as $\mathbf{D}_{ii} = \sum_{j=1}^N \mathbf{W}_{ij}$. The normalized Laplacian matrix is then $\bar{\mathbf{L}} = \mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2}$, where $\mathbf{L} = \mathbf{D} - \mathbf{W}$ is the Laplacian matrix. By decomposing $\bar{\mathbf{L}}$, we obtain the eigenvectors $\{\vec{f}_1, \ldots, \vec{f}_d\} \subseteq \mathbb{R}^N$ corresponding to the smallest d eigenvalues $\{0 < \lambda_1 \leq \ldots \leq \lambda_d\}$. The embedding matrix, providing the low-dimensional representation of the data, is $\mathbf{F} = [\vec{f}_1, \vec{f}_2, \ldots, \vec{f}_d] \in \mathbb{R}^{N\times d}$.

5.2 HALLUCINATION CLUSTER DETECTION

Let $\mathbf{f}_i \in \mathbb{R}^d$ be the i-th row of $\mathbf{F}$, representing the embedding vector of each $\mathbf{h}_i \in \mathcal{S}$. We define the clustering dataset as $\mathcal{C} = \{\mathbf{f}_1, \cdots, \mathbf{f}_N\}$. We can then apply classic clustering algorithms such as Birch Zhang et al. (1996), K-means Likas et al. (2003), Agglomerative Clustering Müllner (2011), and Spectral Clustering Ng et al. (2001) to perform unsupervised clustering on $\mathcal{C}$. This process yields cluster labels $\{\tilde{y}_i\}_{i=1}^N$ for each data point. We will further explore the impact of different clustering algorithms on hallucination detection performance in Section 6.3.

5.3 HALLUCINATION DETECTOR

Based on the cluster labels $\{\tilde{y}_i\}_{i=1}^N$, we can use the training samples $\{(\mathbf{h}_i, \tilde{y}_i)\}_{i=1}^N$ to train a binary classifier $G_\theta : \mathcal{H} \to \{0, 1\}$ for hallucination detection, which is defined as

$$G_\theta(\mathbf{h}) = \text{sigmoid}\left(F_{\mathcal{M}}\left(\sigma(\mathbf{h}\mathbf{W}_1 + \mathbf{b}_1)\right)\mathbf{W}_2 + \mathbf{b}_2\right). \tag{9}$$

Here, σ denotes the ReLU activation function, $\mathbf{W}_1 \in \mathbb{R}^{D\times d}$ and $\mathbf{W}_2 \in \mathbb{R}^{d\times 1}$ are the weight parameters, and $\mathbf{b}_1$ and $\mathbf{b}_2$ are the bias parameters. The module $F_{\mathcal{M}}$, proposed by Yao et al. (2024b), is a manifold fitting sub-module that projects input data near the manifold where a given target dataset resides. The clustering dataset $\mathcal{C} = \{\mathbf{f}_i\}_{i=1}^N$, obtained from manifold embedding, is used as the target dataset (see Figure 1). For more details about $F_{\mathcal{M}}$, please refer to Section A.3.

6 EXPIREMENTS

This section presents the experimental setup, key conclusions, and results from comparative and ablation studies.

6.1 SETUP

Datasets. We used four generative question-answering (QA) datasets: TruthfulQA Lin et al. (2022a), TriviaQA Joshi et al. (2017), NQ Open Kwiatkowski et al. (2019), and SciQ Welbl et al. (2017).TruthfulQA is open-book conversational QA datasets with 817 QA pairs. TriviaQA is a closed-book QA

Table 1: Comparison results (AUROC%) with various competitive hallucination detection algorithms across different datasets and models. "Single sampling" indicates whether the approach requires multiple generations during inference. Both Haloscope and our method report the mean and standard deviation over 5 independent repeated experiments. The results of the remaining methods are cited from Park et al. (2025). **Bold** numbers are superior results.

Model	Method	Single sampling	TruthfulQA	TriviaQA	SciQ	NQ Open
LLaMA-3.1-8b	Perplexity	✓	71.4	76.3	52.6	50.3
	LN-Entropy	×	62.5	55.8	57.6	52.7
	Semantic Entropy	×	59.4	68.7	68.2	60.7
	Lexical Similarity	×	49.1	71.0	61.0	60.9
	EigenScore	×	45.3	69.1	59.6	56.7
	SelfCKGPT	×	57.0	80.2	67.9	60.0
	Verbalize	✓	50.4	51.1	53.4	50.7
	Self-evaluation	✓	67.8	50.9	54.6	52.2
	CCS	✓	66.4	60.1	77.1	62.6
	TSV	✓	84.2	84.0	85.8	76.1
	HaloScope	✓	$83.27^{\pm 2.97}$	$83.52^{\pm 5.31}$	$80.19^{\pm 2.37}$	$71.48^{\pm 6.28}$
	HDME (Ours)	✓	$\mathbf{86.56}^{\pm 3.75}$	$\mathbf{85.53}^{\pm 1.09}$	$\mathbf{86.15}^{\pm 3.51}$	$\mathbf{82.00}^{\pm 2.50}$
Qwen-2.5-7b	Perplexity	✓	65.1	50.2	53.4	51.2
	LN-Entropy	×	66.7	51.1	52.4	54.3
	Semantic Entropy	×	66.1	58.7	65.9	65.3
	Lexical Similarity	×	49.0	63.1	62.2	61.2
	EigenScore	×	53.7	61.3	63.2	57.4
	SelfCKGPT	×	61.7	62.3	58.6	63.4
	Verbalize	✓	60.0	54.3	51.2	51.2
	Self-evaluation	✓	73.7	50.9	53.8	52.4
	CCS	✓	67.9	53.0	51.9	51.2
	TSV	✓	87.3	79.8	**82.0**	73.8
	HaloScope	✓	$83.43^{\pm 3.15}$	$72.31^{\pm 7.68}$	$75.29^{\pm 3.19}$	$79.35^{\pm 4.34}$
	HDME (Ours)	✓	$\mathbf{88.65}^{\pm 2.08}$	$\mathbf{81.29}^{\pm 0.92}$	$81.01^{\pm 2.53}$	$\mathbf{89.95}^{\pm 3.46}$

dataset, and we used the deduplicated validation subset containing 9,960 QA pairs. We used a 3:1 training-to-test set ratio and extracted 100 samples from the training set as a validation set for each dataset. Dataset $\mathcal{N}$ was generated using a greedy sampling strategy, as described in Theorem 3.1.

Models. Since our method requires access to hidden layer representations, our experiments are conducted on two series of open-source models: LLaMA-3.1-8b Dubey et al. (2024), and Qwen-2.5-7b Qwen et al. (2025). We deploy these models on a local server using pre-trained weights and perform zero-shot inference with frozen parameters.

Baselines. Our primary baseline is Haloscope Du et al. (2024), a pioneering method that uses linear matrix decomposition on latent space representations for unsupervised hallucination detection. More span several categories: (1) uncertainty-based Perplexity Ren et al. (2022), LN-entropy Malinin & Gales (2021), Semantic Entropy Chen et al. (2024a); (2) consistency-based Lexical Similarity Lin et al. (2023), SelfCKGPT Manakul et al. (2023), EigenScore Chen et al. (2024a); (3) prompting-based Verbalize Lin et al. (2022b), Self-evaluation Kadavath et al. (2022); (4) knowledge discovery CCS Burns et al. (2023); and (5) latent space steering TSV Park et al. (2025).

Evaluation. Following Du et al. (2024) and Lin et al. (2022a), we generate ground-truth labels by computing BLEURT Sellam et al. (2020) similarity scores between generated content and its ground truth. As hallucination detection is a binary classification task, we evaluate all methods using the Area Under the Receiver Operating Characteristic (AUROC) curve, consistent with prior work Du et al. (2024); Manakul et al. (2023); Kuhn et al. (2023). To demonstrate robustness, we also report results using Rouge-L Lin (2004) as the similarity measure (see Section A.5.3).

Implementation details. For the classifier G_θ defined in Equation (9), we initialize the parameters r_0, r_1, and r_2 in the $F_\mathcal{M}$ submodule with their default values: $r_0 = r_2 = 0.05$ and $r_1 = 0.1$, and use a learnable strategy. The classifier in Haloscope uses the default settings reported in their paper. We use the Adam optimizer andJimmy Ba (2015) with the following parameters: a maximum of 50 iterations, a batch size of 512, an initial learning rate of 0.05, and a cosine learning rate decay with

an initial decay rate of 0.003. Following Du et al. (2024), we use the validation set to determine the extraction layer l for the representation vectors and the dimension d of the manifold embedding. All experiments were conducted using eight NVIDIA A6000 GPUs.

6.2 Main Results

As detailed in Table 1, our method, HDME, outperforms competing hallucination detectors. Across five independent experiments on two open-source models and four datasets, the mean performance of HDME surpasses the Haloscope, notably achieving an 10.52% improvement on NQ Open. Furthermore, HDME exhibits lower standard deviations, indicating greater robustness to random seed initialization. Computationally, HDME is as efficient as Haloscope, as neither requires multiple sampling generations, unlike methods such as Perplexity Ren et al. (2022) and Self-evaluation Kadavath et al. (2022). However, HDME offers significant advantages over Haloscope: it automates label generation $\{\tilde{y}_i\}_{i=1}^N$ through clustering, which eliminates manual thresholding T and reduces hyperparameter dependency. Additionally, its classifier's MLP dimension is substantially smaller ($d \leq 10$ vs. 1024), drastically lowering training costs.

6.3 Ablation Study.

The impact of the feature extraction layer l on HDME. As shown in Figure 3(a), we conduct hallucination detection using representation vectors extracted from different layers of Qwen-2.5-14b, with all experimental settings consistent with the main experiment. We observe that HDME's performance initially increases with the number of layers, peaks between layers 10 and 20, and then decreases. This trend aligns with Haloscope's performance, confirming the findings of Azaria & Mitchell (2023); Chen et al. (2024a) that intermediate layer representations are more effective for downstream tasks.

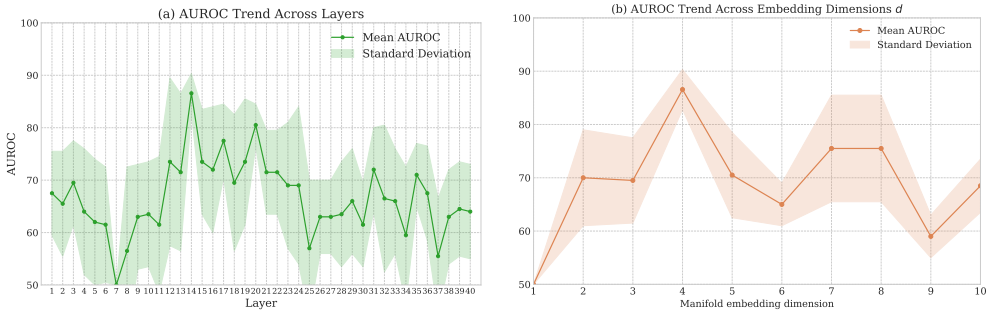


Figure 3: Ablation experiments on l and d. (a) The impact of different feature extraction layers l on HDME performance. (b) The variation of HDME performance with manifold embedding dimension d. All experiments are conducted on Qwen-2.5-14b and TruthfulQA.

Ablation on manifold embedding dimension d. As shown in Figure 3(b), we explore the impact of manifold embedding dimension d on HDME performance using the Qwen-2.5-14b model, with the feature extraction layer fixed at $l = 14$ (the optimal setting for Qwen-2.5-14b, see Figure 3(a)). We find that embedding the high-dimensional raw data into a 3-5 dimensional subspace yields better results, supporting the validity of Theorem 5.1.

Exploration of the effectiveness of the manifold modules in HDME. We performed an ablation study to assess the two manifold modules in our HDME method: the projection module $E_{\mathcal{Z}}$ (Equation (8)) and the fitting submodule $F_{\mathcal{M}}$ within the detector G_θ. As shown in Table 2, $E_{\mathcal{Z}}$ primarily increases the mean AUROC, while $F_{\mathcal{M}}$ reduces its standard deviation, indicating that they respectively enhance performance and robustness. The performance gain from $E_{\mathcal{Z}}$ stems from its ability to increase the initial separation between hallucinated and factual samples. The robustness from $F_{\mathcal{M}}$ is achieved by projecting embeddings onto the learned manifold $\hat{\mathcal{M}}$, which amplifies sample separability for the final MLP classifier in G_θ. Using both modules yields the best results. Notably, our classifier's MLP dimension of $d \leq 10$ is significantly smaller than the 1024 used by Haloscope, substantially reducing training costs.

Ablation study on the selection of clustering algorithms. Since the hallucination classifier is trained on clustering-derived labels, its performance is contingent on the clustering algorithm's accuracy. We ablated four common clustering algorithms, using default sklearn parameters except for Birch (threshold=0.01) and SpectralClustering (affinity='rbf'). The AUROC results in Table 2 demonstrate that spectral clustering provides superior performance, achieving the highest mean and lowest standard deviation. We attribute this to its ability to leverage spectral information that the $E_{\mathcal{Z}}$ module preserves when embedding the data into the low-dimensional manifold.

Table 2: Performance with different manifold module designs and clustering algorithms.

Clustering algorithm	Module design		TruthfulQA	TriviaQA	TruthfulQA	TriviaQA
	$E_{\mathcal{Z}}$	$F_{\mathcal{M}}$	LLaMA-3.1-8b		Qwen-2.5-7b	
Spectral	×	×	$70.24^{\pm 3.23}$	$78.39^{\pm 0.95}$	$67.15^{\pm 1.21}$	$71.19^{\pm 1.32}$
	×	✓	$68.48^{\pm 1.73}$	$83.12^{\pm 0.44}$	$75.69^{\pm 4.11}$	$75.68^{\pm 0.98}$
	✓	×	$78.16^{\pm 4.60}$	$83.35^{\pm 0.67}$	$80.06^{\pm 3.42}$	$79.67^{\pm 1.06}$
Spectral	✓	✓	$\mathbf{86.56}^{\pm 3.75}$	$\mathbf{85.53}^{\pm 1.09}$	$\mathbf{88.65}^{\pm 2.08}$	$\mathbf{81.29}^{\pm 0.92}$
Agglomerative	✓	✓	$74.65^{\pm 5.74}$	$84.34^{\pm 2.03}$	$62.09^{\pm 6.87}$	$78.73^{\pm 1.02}$
Birch	✓	✓	$74.72^{\pm 5.23}$	$84.74^{\pm 1.59}$	$53.78^{\pm 4.58}$	$66.43^{\pm 1.38}$
K-means	✓	✓	$76.50^{\pm 5.94}$	$83.68^{\pm 1.62}$	$76.16^{\pm 4.30}$	$80.27^{\pm 1.53}$

Comparison with supervised training on labeled data. Similar to Du et al. (2024), we explored the performance gap between our method, HDME, and the upper-performance limit by comparing our results with those obtained through fully supervised training. As illustrated in Figure 4, our method demonstrates superior performance compared to Haloscope, achieving detection accuracy that more closely approximates supervised approaches while exhibiting enhanced robustness.

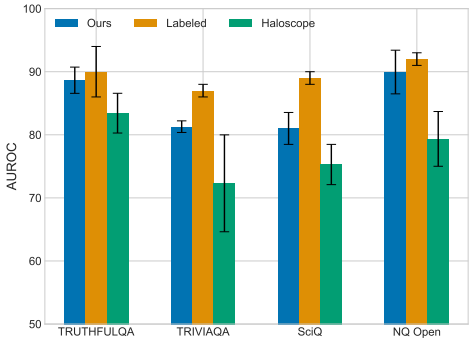


Figure 4: Comparison of performance on Qwen-2.5-7b and TruthfulQA with fully supervised training.

Training Time Comparison. We compare the training time of HDME and Haloscope under identical settings, including computational resources and search ranges for shared hyperparameters (Table 3). S̈inglëtraining denotes a 50-iteration run, while f̈ull̈training includes the complete grid search. The hyperparameter space for HDME consists of layers l and manifold dimension d. Haloscope's search space is larger, as it also includes a threshold parameter T. Results in Table 3 show that while HDME is marginally slower in single training due to its manifold submodule, it is substantially more efficient in full training. This advantage stems from HDME's use of clustering, which circumvents the costly search for the threshold T required by Haloscope.

Table 3: Training time (s) comparison between HDME and Haloscope.

Training stage	Method	TruthfulQA	TriviaQA	TruthfulQA	TriviaQA
		Qwen-2.5-7b		LLaMA-3.1-8b	
single	Haloscope	0.37	1.87	0.64	3.13
	HDME(Ours)	0.42	3.42	0.81	4.01
full	Haloscope	725.05	7920.90	1409.43	9006.4
	HDME(Ours)	172.68	5036.82	557.06	6742.82

7 Conclusion

In summary, our proposed HDME method aims to tackle this hallucination issue. By introducing HARs and TARs, we deepen the understanding of the non-linearly separable distribution of hallucinated and truthful samples in LLMs' latent space, which was fundamental for HDME's development. The HDME framework, with manifold embedding and clustering, effectively overcomes high-dimensional and noisy data challenges, enhancing hallucination detection for unlabeled data. Experimental results clearly show HDME's superiority over other methods, with improved mean AUROC and reduced standard deviation. Also, we conduct in-depth ablation studies of HDME for future improvements.

Ethics Statements

This research does not raise any ethical concerns. The study exclusively involved the analysis of publicly available data sets and published literature, which did not contain any personally identifiable information. No human participants, animals, or sensitive data were involved in this research. All sources are properly cited in accordance with academic standards. The authors confirm that this work was conducted in accordance with the principles of academic integrity and research ethics.

Reproducibility Statement

We ensure full reproducibility by publicly releasing all relevant materials of codes and data resources. All results were generated using fixed computational resources detailed in Section 6 and Appendices. This enables independent verification of all findings.

References

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

Diederik P. Kingma andJimmy Ba. Adam: A method for stochastic optimization. In *In International Conference on Learning Representations*, 2015.

Amos Azaria and Tom Mitchell. The internal state of an llm knows when it's lying. *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

Mukund Balasubramanian and Eric L Schwartz. The isomap algorithm and topological stability. *Science*, 295(5552):7–7, 2002.

Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 15(6):1373–1396, 2003.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. Discovering latent knowledge in language models without supervision. *International Conference on Learning Representations,*, 2023.

Chao Chen, Kai Liu, Ze Chen, Yi Gu, Yue Wu, Mingyuan Tao, Zhihang Fu, and Jieping Ye. Inside: Llms' internal states retain the power of hallucination detection. *In International Conference on Learning Representations*, 2024a.

Xiang Chen, Duanzheng Song, Honghao Gui, Chengxi Wang, Ningyu Zhang, Jiang Yong, Fei Huang, Chengfei Lv, Dan Zhang, and Huajun Chen. Factchd: Benchmarking fact-conflicting hallucination detection. In *Proceedings of the 33rd International Joint Conference on Artificial Intelligence*, 2024b.

Yuyan Chen, Qiang Fu, Yichen Yuan, Zhihao Wen, Ge Fan, Dayiheng Liu, Dongmei Zhang, Zhixu Li, and Yanghua Xiao. Hallucination detection: Robustly discerning reliable answers in large language models. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pp. 245–255, 2023.

Jonathan H Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. Tydi qa: A benchmark for information-seeking question answering in ty pologically di verse languages. *Transactions of the Association for Computational Linguistics*, 8: 454–470, 2020.

Jonathan Crabbé and Mihaela van der Schaar. Concept activation regions: A generalized framework for concept-based explanations. *Advances in Neural Information Processing Systems*, 35:2590–2607, 2022.

Mark L Davison and Stephen G Sireci. Multidimensional scaling. In *Handbook of applied multivariate statistics and mathematical modeling*, pp. 323–352. Elsevier, 2000.

Xuefeng Du, Chaowei Xiao, and Yixuan Li. Haloscope: Harnessing unlabeled llm generations for hallucination detection. In *Advances in Neural Information Processing Systems*, 2024.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Charles Fefferman, Sergei Ivanov, Yaroslav Kurylev, Matti Lassas, and Hariharan Narayanan. Fitting a putative manifold to noisy data. In *Conference On Learning Theory*, pp. 688–720. PMLR, 2018.

Charles Fefferman, Sergei Ivanov, Matti Lassas, and Hariharan Narayanan. Fitting a manifold of large reach to noisy data. *Journal of Topology and Analysis*, pp. 1–82, 2023.

Jinrong He, Lixin Ding, Lei Jiang, Zhaokui Li, and Qinghui Hu. Intrinsic dimensionality estimation based on manifold assumption. *Journal of Visual Communication and Image Representation*, 25 (5):740–747, 2014.

Thomas Hofmann, Bernhard Schölkopf, and Alexander J Smola. Kernel methods in machine learning. *The Annals of Statistics*, 36(3):1171–1220, 2008.

Peter J Huber. Robust estimation of a location parameter. In *Breakthroughs in statistics: Methodology and distribution*, pp. 492–518. Springer, 1992.

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

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pp. 1601—-1611, 2017.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Haoqiang Kang, Terra Blevins, and Luke Zettlemoyer. Comparing hallucination detection metrics for multilingual generation. *arXiv preprint arXiv:2402.10496*, 2024.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. *In International Conference on Learning Representations*, 2023.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466, 2019.

Aristidis Likas, Nikos Vlassis, and Jakob J Verbeek. The global k-means clustering algorithm. *Pattern recognition*, 36(2):451–461, 2003.

Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pp. 74–81, 2004.

Stephanie Lin, Jacob Hilton, and Owain Evans. Truthfulqa: Measuring how models mimic human falsehoods. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, 2022a.

Stephanie Lin, Jacob Hilton, and Owain Evans. Teaching models to express their uncertainty in words. *Transactions on Machine Learning Research*, 2022b.

Tong Lin and Hongbin Zha. Riemannian manifold learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(5):796–809, 2008a.

Tong Lin and Hongbin Zha. Riemannian manifold learning. *IEEE transactions on pattern analysis and machine intelligence*, 30(5):796–809, 2008b.

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

Tianyu Liu, Yizhe Zhang, Chris Brockett, Yi Mao, Zhifang Sui, Weizhu Chen, and Bill Dolan. A token-level reference-free hallucination detection benchmark for free-form text generation. *arXiv preprint arXiv:2104.08704*, 2021.

Junliang Luo, Tianyu Li, Di Wu, Michael Jenkin, Steve Liu, and Gregory Dudek. Hallucination detection and hallucination mitigation: An investigation. *arXiv preprint arXiv:2401.08358*, 2024.

Andrzej Maćkiewicz and Waldemar Ratajczak. Principal components analysis (pca). *Computers & Geosciences*, 19(3):303–342, 1993.

Andrey Malinin and Mark Gales. Uncertainty estimation in autoregressive structured prediction. *In International Conference on Learning Representations*, 2021.

Potsawee Manakul, Adian Liusie, and Mark JF Gales. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

Marina Meilă and Hanyu Zhang. Manifold learning: What, how, and why. *Annual Review of Statistics and Its Application*, 11, 2024.

Daniel Müllner. Modern hierarchical, agglomerative clustering algorithms. *arXiv preprint arXiv:1109.2378*, 2011.

Andrew Ng, Michael Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, 14, 2001.

Seongheon Park, Xuefeng Du, Min-Hsuan Yeh, Haobo Wang, and Yixuan Li. Steer llm latents for hallucination detection. *arXiv preprint arXiv:2503.01917*, 2025.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL `https://arxiv.org/abs/2412.15115`.

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

Jie Ren, Jiaming Luo, Yao Zhao, Kundan Krishna, Mohammad Saleh, Balaji Lakshminarayanan, and Peter J Liu. Out-of-distribution detection and selective generation for conditional language models. In *In The Eleventh International Conference on Learning Representations*, 2022.

Sam T Roweis and Lawrence K Saul. Nonlinear dimensionality reduction by locally linear embedding. *science*, 290(5500):2323–2326, 2000.

Thibault Sellam, Dipanjan Das, and Ankur P Parikh. Bleurt: Learning robust metrics for text generation. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 7881–7892, 2020.

Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence*, 22(8):888–905, 2000.

Weihang Su, Changyue Wang, Qingyao Ai, Yiran Hu, Zhijing Wu, Yujia Zhou, and Yiqun Liu. Unsupervised real-time hallucination detection based on the internal states of large language models. *arXiv preprint arXiv:2403.06448*, 2024.

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

Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.

Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(3):395–416, 2007.

Johannes Welbl, Nelson F Liu, and Matt Gardner. Crowdsourcing multiple choice science questions. *arXiv preprint arXiv:1707.06209*, 2017.

Zhigang Yao, Jiaji Su, Bingjie Li, and Shing-Tung Yau. Manifold fitting. *arXiv preprint arXiv:2304.07680*, 2023.

Zhigang Yao, Bingjie Li, Yukun Lu, and Shing-Tung Yau. Single-cell analysis via manifold fitting: A framework for rna clustering and beyond. *Proceedings of the National Academy of Sciences*, 121 (37):e2400002121, 2024a.

Zhigang Yao, Jiaji Su, and Shing-Tung Yau. Manifold fitting with cyclegan. *Proceedings of the National Academy of Sciences*, 121(5):e2311436121, 2024b.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022.

Tian Zhang, Raghu Ramakrishnan, and Miron Livny. Birch: an efficient data clustering method for very large databases. *ACM sigmod record*, 25(2):103–114, 1996.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 2023.

A Appendix / supplemental material

A.1 Transformer decoder block.

Considering an L-layer, H-head LLM with a causal decoder structure, where the model dimension is D and the attention dimension is d_k, the l-th layer transformer decoder block can be defined as:

$$\begin{aligned} TF^l\left(\mathbf{H}^{l-1}\right) &= \Pi\big(\sigma\big(\mathbf{Z}^l\mathbf{W}^l_{\mathrm{f1}}\big)\mathbf{W}^l_{\mathrm{f2}} + \mathbf{Z}^l\big), \\ \mathbf{Z}^l &= \Pi\big(\sum_{h\in[H]} A^l_h(\mathbf{H}^{l-1})\mathbf{W}^l_{O_h} + \mathbf{H}^{l-1}\big), \end{aligned} \tag{10}$$

here Π denotes the Layer-normlization operator, σ denotes the non-linear activation function and $\mathbf{W}^l_{(\cdot)}$ are the weight matrixes. $A^l_h(\cdot)$ denotes the masked self-attention of the h-th head at the l-the layer, defined as **Msked Self-attention.** Denote $\mathrm{softmax}$ as the row-wise softmax operator, $\mathbf{Q}^l_h = \mathbf{Z}^{l-1}\mathbf{W}^l_{Q_h}$, $\mathbf{K}^l_h = \mathbf{Z}^{l-1}\mathbf{W}^l_{K_h}$, $\mathbf{V}^l_h = \mathbf{Z}^{l-1}\mathbf{W}^l_{V_h}$, we have

$$A^l_h(\mathbf{H}^{l-1}) = \mathrm{softmax}\left(\frac{\mathbf{Q}^l_h(\mathbf{K}^l_h)^\top}{\sqrt{d_k}} + \mathbf{M}\right)\mathbf{V}^l_h, \tag{11}$$

where $\mathrm{softmax}$ denotes the row-wise softmax operator, $\mathbf{Q}^l_h = \mathbf{H}^{l-1}\mathbf{W}^l_{Q_h}$, $\mathbf{K}^l_h = \mathbf{H}^{l-1}\mathbf{W}^l_{K_h}$, $\mathbf{V}^l_h = \mathbf{H}^{l-1}\mathbf{W}^l_{V_h}$. $\mathbf{M} \in \mathbb{R}^{m\times m}$ is a mask matrix defined as

$$\mathbf{M}_{ij} = \begin{cases} 0, & j \le i \\ -\infty, & j > i \end{cases} \tag{12}$$

Denote $\mathbf{h}_j, \mathbf{q}_j, \mathbf{k}_j, \mathbf{v}_j, \mathbf{m}_j$ as the j-th row of $\mathbf{H}, \mathbf{Q}, \mathbf{K}, \mathbf{V}, \mathbf{M}$ respectively (ignoring layer and head parameters), we have:

$$\begin{aligned} A(\mathbf{h}_j) &= \mathrm{softmax}\left(\frac{\mathbf{q}_j\mathbf{K}^\top}{\sqrt{d_k}} + \mathbf{m}_j\right)\mathbf{V} \\ &= \mathrm{softmax}\left(\frac{\mathbf{q}_j\mathbf{k}_1^\top}{\sqrt{d_k}}, \cdots, \frac{\mathbf{q}_j\mathbf{k}_j^\top}{\sqrt{d_k}}, -\infty, \cdots, -\infty\right)\mathbf{V} \\ &= (\mathbf{S}_{j1}, \cdots, \mathbf{S}_{jj}, 0, \cdots, 0)\,\mathbf{V} \\ &= \mathbf{S}_{j1}\mathbf{v}_1 + \cdots + \mathbf{S}_{jj}\mathbf{v}_j, \end{aligned} \tag{13}$$

where $\mathbf{S}_{jk}$ denotes the j-th row and k-th column element of the matrix $\mathrm{softmax}\left(\mathbf{Q}\mathbf{K}^\top/\sqrt{d_k} + \mathbf{M}\right)$. According to Equation (13), each token in the self-attention mechanism can utilize only information from preceding nodes, owing to the application of the mask $\mathbf{M}$.

A.2 Manifold projection.

The manifold projection consists of the following two steps Yao et al. (2024a):

Direction estimation. Given a set of unlabeled data points $\mathcal{S} = \{\mathbf{h}_i \in \mathbb{R}^D\}_{i=1}^N$, the projection direction $F(\mathbf{h}_i)$ of $\mathbf{h}_i$ is defined by

$$F(\mathbf{h}_i) = \frac{1}{|\mathbb{B}_i|}\sum_{\mathbf{h}_j\in\mathbb{B}_i}\mathbf{h}_j, \tag{14}$$

where

$$\mathbb{B}_i = \arg\max_{\mathcal{W}\subset\mathcal{S},|\mathcal{W}|=p}\sum_{\mathbf{h}_j\in\mathcal{W}}\mathrm{SNN}(i,j), \tag{15}$$

and

$$\mathrm{SNN}(i,j) = \left|\mathcal{N}_p(i)\cap\mathcal{N}_p(j)\right|. \tag{16}$$

Here, we denote $\mathcal{N}_p(i)$ as the set of the p nearest neighbors of each vector $\mathbf{h}_i$, determined by a specified metric. The shared nearest neighborhood, $\mathrm{SNN}(i,j)$, is then defined as the intersection of the nearest neighbor sets of $\mathbf{h}_i$ and $\mathbf{h}_j$.

Projection estimation. For any data point $\mathbf{h}_i$, its projection direction can be expressed as $E_{\mathcal{Z}}(\mathbf{h}_i)$. We simplify $J(\mathbf{h}_i)$ by identifying the point of maximum density along the line connecting $\mathbf{h}_i$ and $F(\mathbf{h}_i)$ and using this point as a substitute for the original $E_{\mathcal{Z}}(\mathbf{h}_i)$. Precisely, the simplification is

$$E_{\mathcal{Z}}(\mathbf{h}_i) = \arg\max_{\mathbf{h}_t} \varrho(\mathbf{h}_t), \tag{17}$$

where

$$\mathbf{h}_t = \mathbf{h}_i + t(F(\mathbf{h}_i) - \mathbf{h}_i), \tag{18}$$

and

$$\varrho(\mathbf{h}_i) = \frac{1}{\sum_{\mathbf{h}_j \in \mathbb{B}_i} \|\mathbf{h}_i - \mathbf{h}_j\|_2^2} \tag{19}$$

quantifies the density of $\mathbf{h}_i$, with higher values indicating denser regions.

A.3 Manifold fitting sub-module Yao et al. (2024b).

In this section, we introduce the $F_{\mathcal{M}}$ manifold fitting sub-module for hallucination detection based on the latent space of large language models (LLMs). As illustrated in Figure 1, our target sample set is $\mathcal{C} = \{\mathbf{f}_1, \ldots, \mathbf{f}_N\} \subseteq \mathbb{R}^d$. Assume that $\mathbf{f}_i = z_i + \xi_i$, where $\xi_i \sim \phi_\delta$ is the noise, $z_i \sim \omega$ is the Noise-free sample.We start by making the following assumptions:

Assumption A.1. Our ambient space is a d–dimensional Euclidean space $\mathcal{Y} = \mathbb{R}^d$ with the standard Euclidean norm. The noise distribution ϕ_δ in $\mathcal{Y}$ is a Gaussian distribution on $\mathcal{Y}$, and the density at point ξ can be defined as

$$\phi_\delta(\xi) = \big(\frac{1}{2\pi\delta^2}\big)^{d/2} \exp\big(-\frac{\|\xi\|_2^2}{2\delta^2}\big).$$

There exists a compact d'-dimensional ($d' \leq d$) sub-manifold $\mathcal{M}$ embedded in $\mathcal{Y}$ which is twice-differentiable. The distribution ω on $\mathcal{M}$ is a uniform distribution with respect to d'-dimensional Hausdorff measure. The intrinsic dimension d' and the standard deviation δ of the noise distribution are both known.

Then, for any point $y \in \mathcal{Y}$, we can use the $F_{\mathcal{M}}$ to find its contracted point, calculated as

$$F_{\mathcal{M}}(y) = \sum_i \beta_i(y)\mathbf{f}_i \tag{20}$$

with the weights given by

$$\begin{gathered} w_u(u_i) = \begin{cases} 1, & \|u_i\|_2 \leq \frac{r_2}{2} \\ \left(1 - \left(\frac{2\|u_i\|_2 - r_2}{r_2}\right)^2\right)^k, & \|u_i\|_2 \in \left(\frac{r_2}{2}, r_2\right) \\ 0, & \text{otherwise} \end{cases} \\ w_v(v_i) = \begin{cases} \left(1 - \frac{\|v_i\|_2^2}{r_1^2}\right)^k, & \|v_i\|_2 \leq r_1 \\ 0, & \text{otherwise} \end{cases} \\ \beta_i(y) = w_u(u_i)\, w_v(v_i), \quad \tilde{\beta}(y) = \sum \tilde{\beta}_i(y), \quad \beta_i(y) = \frac{\tilde{\beta}_i(y)}{\tilde{\beta}(y)}, \end{gathered} \tag{21}$$

where

$$u_i = \Pi_y(\mathbf{f}_i - y), \quad v_i = \mathbf{f}_i - y - u_i, \tag{22}$$

where Π_y is a projection matrix defined as

$$\Pi_y = \frac{(\mu_y - y)(\mu_y - y)^T}{\|(\mu_y - y)\|_2^2}, \tag{23}$$

and μ_y denotes the contraction direction of y, which can be defined as

$$\mu_y = \sum_i \alpha_i(y)\mathbf{f}_i, \tag{24}$$

where the weights are defined as

$$\tilde{\alpha}_i(y) = \begin{cases} \left(1 - \frac{\|y - \mathbf{f}_i\|_2^2}{r_0^2}\right)^k, & \|y - \mathbf{f}_i\|_2 \leq r_0 \\ 0, & \text{otherwise} \end{cases}, \quad \tilde{\alpha}(y) = \sum_{i \in I_y} \tilde{\alpha}_i(y), \quad \alpha_i(y) = \frac{\tilde{\alpha}_i(y)}{\tilde{\alpha}(y)}. \tag{25}$$

A.4 HDME FRAMEWORK.

Algorithm 1 HDME Framework

Input: a set of prompts $\{\mathbf{x}_{prompt_1}, \cdots, \mathbf{x}_{prompt_N}\}$
Step 1: Obtain the unlabeled sample set $\mathcal{N}$ as defined in Equation (5) and the corresponding representation set $\mathcal{S}$ as defined in Equation (6).
Step 2: Perform manifold fitting on the D-dimensional dataset $\mathcal{S}$ to obtain the d-dimensional embedding set $\mathcal{C}$.
Step 3: Cluster the dataset $\mathcal{C}$ to obtain the corresponding cluster labels $\{\tilde{y}_i\}_{i=1}^N$, where $d \ll D$.
Step 4: Use the representation set $\mathcal{S}$ and the cluster labels $\{\tilde{y}_i\}_{i=1}^N$ to train a binary classifier G_θ as defined in Equation (9).
Return: a hallucination detector $G_{\hat{\theta}} : \mathcal{H} \to \{0, 1\}$

A.5 MORE EXPERIMENTS.

A.5.1 OTHER MODELS AND DATASET.

Datasets. We utilized four generative question-answering (QA) datasets: CoQA Reddy et al. (2019), TruthfulQA Lin et al. (2022a), TriviaQA Joshi et al. (2017), and TydiQA Clark et al. (2020).

Models. We utilized three open-source model series: OPT-6.7B and 13B Zhang et al. (2022), Llama2-chat-7B and 13B Touvron et al. (2023), and Llama3-8B Dubey et al. (2024).

Table 4: Main results on other models and dataset. All values are AUROC %. Both Haloscope and our method report the mean and standard deviation over 5 independent repeated experiments. The results of the remaining methods are cited from Du et al. (2024). **Bold** numbers are superior results.

Model	Method	Single sampling	TruthfulQA	TriviaQA	CoQA	TydiQA-GP
Llama2-7b	Perplexity	×	56.77	72.13	69.45	78.45
	LN-Entropy	×	61.51	70.91	72.96	76.27
	Semantic Entropy	×	63.50	73.21	63.21	73.89
	Lexical Similarity	×	55.69	71.30	66.50	77.06
	EigenScore	×	61.45	68.45	73.22	79.27
	SelfCKGPT	×	52.95	73.22	73.24	77.80
	Verbalize	✓	53.04	67.37	71.32	49.47
	Self-evaluation	✓	51.81	55.68	48.23	48.36
	CCS	×	54.27	51.11	51.48	47.59
	CCS*	✓	67.95	63.60	51.32	50.38
	HaloScope	✓	$82.04^{\pm 3.73}$	$82.31^{\pm 4.49}$	$75.08^{\pm 1.47}$	$84.75^{\pm 10.03}$
	HDME (Ours)	✓	$\mathbf{84.79}^{\pm 2.65}$	$\mathbf{83.98}^{\pm 1.06}$	$\mathbf{79.25}^{\pm 1.23}$	$\mathbf{96.43}^{\pm 0.50}$
OPT-6.7b	Perplexity	×	59.13	69.51	70.21	63.50
	LN-Entropy	×	54.42	71.42	71.23	62.70
	Semantic Entropy	×	52.44	70.21	71.02	62.70
	Lexical Similarity	×	49.13	71.07	66.56	70.36
	EigenScore	×	51.43	70.07	64.43	70.14
	SelfCKGPT	×	50.17	71.49	70.24	63.97
	Verbalize	✓	51.30	50.92	47.29	52.59
	Self-evaluation	✓	51.00	53.02	47.29	50.92
	CCS	×	53.00	51.11	51.48	47.59
	CCS*	✓	63.91	53.89	57.50	53.92
	HaloScope	✓	$74.86^{\pm 6.02}$	$75.44^{\pm 10.87}$	$73.81^{\pm 2.34}$	$83.20^{\pm 0.84}$
	HDME (Ours)	✓	$\mathbf{80.64}^{\pm 4.08}$	$\mathbf{80.27}^{\pm 0.80}$	$\mathbf{75.52}^{\pm 0.99}$	$\mathbf{86.49}^{\pm 0.56}$

A.5.2 SCALABILITY TO MODEL SIZE AND TYPE.

To further evaluate the effectiveness of HDME, we compared our method with HaloScope on two larger models, Llama2-13b-chat and OPT-13b. The results in Table 5 indicate that our method consistently outperforms and is more stable than HaloScope, demonstrating the excellent scalability

of HDME. Additionally, experimental results on the latest model, Llama3-8b, further confirm the scalability of HDME.

Table 5: Hallucination detection performance on larger LLMs.

Model	Method	TruthfulQA	TriviaQA	CoQA	TydiQA-GP
Llama2-13b	HaloScope	$72.45^{\pm 5.51}$	$89.22^{\pm 6.15}$	$70.12^{\pm 8.36}$	$71.05^{\pm 11.63}$
	HDME (Ours)	$\mathbf{78.39}^{\pm 2.67}$	$\mathbf{91.06}^{\pm 1.39}$	$\mathbf{76.89}^{\pm 3.37}$	$\mathbf{84.99}^{\pm 1.92}$
OPT-13b	HaloScope	$85.20^{\pm 3.14}$	$75.68^{\pm 4.95}$	$74.21^{\pm 3.62}$	$88.63^{\pm 0.94}$
	HDME (Ours)	$\mathbf{85.39}^{\pm 3.66}$	$\mathbf{75.88}^{\pm 1.28}$	$\mathbf{75.76}^{\pm 4.68}$	$\mathbf{89.15}^{\pm 0.68}$
Llama3-8b	HaloScope	$68.28^{\pm 5.18}$	$87.29^{\pm 1.13}$	$65.82^{\pm 6.38}$	$77.25^{\pm 5.48}$
	HDME (Ours)	$\mathbf{75.61}^{\pm 2.97}$	$\mathbf{89.60}^{\pm 0.62}$	$\mathbf{73.26}^{\pm 1.69}$	$\mathbf{82.79}^{\pm 4.48}$

A.5.3 Results with Rouge-L.

To further validate the effectiveness of the HDME framework, we used another metric, Rouge-L Lin (2004), to generate true labels and set the threshold to 0.5. We conducted ablation experiments on three different models: Llama2-7b-chat, OPT-6.7b, and Llama3-8b. As shown in the results in Table 6, our method remains effective.

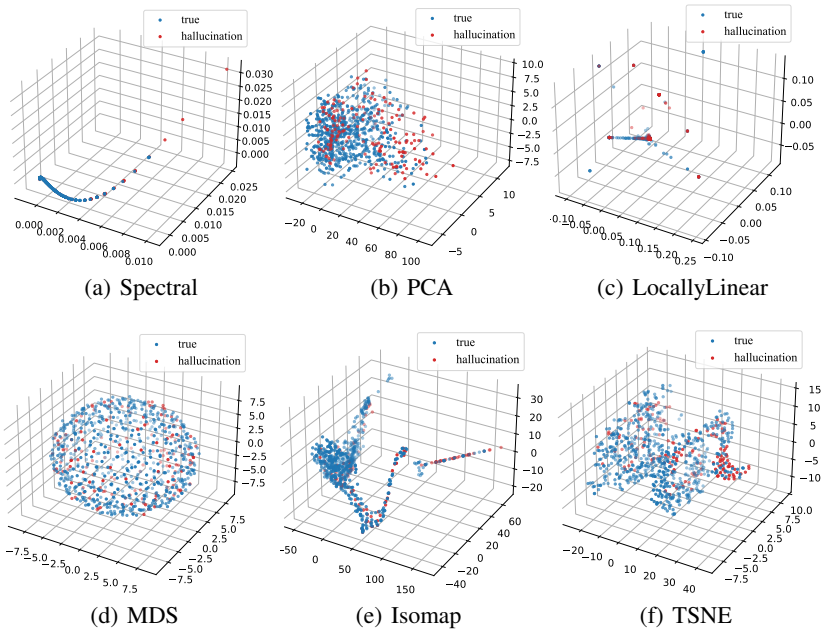
Table 6: Main results with Rouge-L metric. All values are AUROC %. Both Haloscope and our method report the mean and standard deviation over 5 independent repeated experiments. The results of the remaining methods are cited from Du et al. (2024). **Bold** numbers are superior results.

Model	Method	Single sampling	TruthfulQA	TydiQA-GP
Llama2-7b	Perplexity	×	42.62	75.32
	LN-Entropy	×	44.77	73.90
	Semantic Entropy	×	47.01	71.27
	Lexical Similarity	×	67.78	45.63
	EigenScore	×	67.31	47.90
	SelfCKGPT	×	54.05	49.96
	Verbalize	✓	53.71	55.29
	Self-evaluation	✓	55.96	51.04
	CCS	×	59.07	71.62
	CCS*	✓	60.12	77.35
	HaloScope	✓	$80.70^{\pm 4.99}$	$73.36^{\pm 9.91}$
	HDME (Ours)	✓	$\mathbf{81.32}^{\pm 3.30}$	$\mathbf{76.08}^{\pm 1.38}$
OPT-6.7b	HaloScope	✓	$78.31^{\pm 2.34}$	$80.19^{\pm 1.25}$
	HDME (Ours)	✓	$\mathbf{79.19}^{\pm 3.17}$	$\mathbf{81.03}^{\pm 0.70}$

A.6 Analysis of latent space sample distribution.

In this section, we analyze the latent space sample distribution using various manifold embedding methods, including Spectral Embedding Ng et al. (2001), PCA Embedding Maćkiewicz & Ratajczak (1993), Locally Linear Embedding Roweis & Saul (2000), MDS Embedding Davison & Sireci (2000), Isomap Embedding Balasubramanian & Schwartz (2002), and TSNE Embedding Van der Maaten & Hinton (2008). Among these, only PCA is a linear method, while the others are nonlinear.

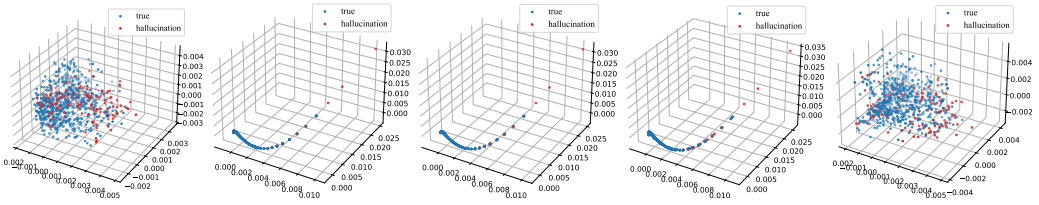
As shown in Figure 5, nonlinear embeddings can project samples onto a manifold structure, where both hallucinated and real samples exhibit clustering effects. In contrast, PCA shows a weaker clustering effect and no clear manifold structure, indicating that the original data is nonlinear and linear methods cannot effectively capture the data's structural information. This supports the validity of Theorem 4.4 and Theorem 5.1. Among the nonlinear methods, spectral embedding performs best in facilitating sample clustering, which is why we chose it as the HDME embedding method. However, the choice of embedding method may depend on the specific model and dataset. While we recommend spectral embedding by default, selecting the most suitable method based on the model and dataset is advisable.



(a) Spectral (b) PCA (c) LocallyLinear

(d) MDS (e) Isomap (f) TSNE

Figure 5: Latent space sample distribution after using different manifold embedding methods. All methods are visualized on Llama2-7b-chat using the TruthfulQA dataset, with the number of feature extraction layers fixed at $l = 16$ and the embedding dimension fixed at $d = 3$.



(a) Layer 1 (b) Layer 8 (c) Layer 16 (d) Layer 24 (e) Layer 32

Figure 6: Using spectral embedding for visualization on Llama2-7b-chat with the TruthfulQA dataset, the embedding dimension is fixed at $d = 3$, and the number of feature extraction layers is set to 1, 8, 16, 24, and 32, respectively.

As shown in Figure 6, the distribution of embedded samples varies with different feature extraction layers. In the first layer, the model has not fully learned the contextual information, resulting in poor performance. In the last layer (32nd layer), the model is overly confident in its context encoding, also leading to poor performance. The best encoding performance is achieved in the middle layers. This finding is consistent with our ablation experiments on the extraction layer number l using HDME.

A.7 LIMITATIONS.

While HDME advances hallucination detection through systematic integration of manifold learning with clustering mechanisms, and demonstrates measurable performance improvements over existing baselines in open-environment settings, we identify four key directions for future research that reflect both our methodological choices and inherent constraints of current detection paradigms:

(1) Theoretical-Experimental Balance: The proposed HARs/TARs conceptual framework provides valuable visual evidence for latent space analysis (Theorem 4.4), though formal theoretical grounding remains challenging given the complex nature of modern LLM architectures. This limitation,

common in interpretability research, points to the need for new mathematical tools to bridge empirical observations with theoretical models.

(2) Model Accessibility Constraints: Like Haloscope, HDME currently requires access to model internals, limiting application to open-source architectures. This reflects a broader challenge in the field that warrants attention as the community develops more sophisticated black-box detection techniques.

B DISCLOSURE OF GENERATIVE AI USAGE

GenAI tools were used during the editing (e.g., grammar, spelling, word choice). And the authors are fully accountable for the content.