Beyond Next Token Probabilities: Learnable, Fast Detection of Hallucinations and Data Contamination on LLM Output Distributions

Guy Bar-Shalom[*,1], **Fabrizio Frasca**[*,1], **Derek Lim**[2,7], **Yoav Gelberg**[1,6], **Yftah Ziser**[3,4], **Ran El-Yaniv**[1,4], **Gal Chechik**[4,5], **Haggai Maron**[1,4]

[1]Technion
[2]Open AI
[3]University of Groningen
[4]Nvidia
[5]Bar-Ilan University
[6]Oxford
[7]MIT
guybs99@gmail.com, fabriziof@campus.technion.ac.il

Abstract

The automated detection of hallucinations and training data contamination is pivotal to the safe deployment of Large Language Models (LLMs). These tasks are particularly challenging in settings where no access to model internals is available. Current approaches in this setup typically leverage only the probabilities of actual tokens in the text, relying on simple task-specific heuristics. Crucially, they overlook the information contained in the full sequence of next-token probability distributions. We propose to go beyond hand-crafted decision rules by learning directly from the complete observable output of LLMs — consisting not only of next-token probabilities, but also the full sequence of next-token distributions. We refer to this as the LLM Output Signature (LOS), and treat it as a reference data type for detecting hallucinations and data contamination. To that end, we introduce LOS-Net, a lightweight attention-based architecture trained on an efficient encoding of the LOS, which can provably approximate a broad class of existing techniques for both tasks. Empirically, LOS-Net achieves superior performance across diverse benchmarks and LLMs, while maintaining extremely low detection latency. Furthermore, it demonstrates promising transfer capabilities across datasets and LLMs.

Code — https://github.com/BarSGuy/Beyond-next-token-probabilities

Extended version — https://arxiv.org/abs/2503.14043

1 Introduction

As the remarkable capabilities of LLMs continue to drive their expanding range of applications, detecting hallucinations (Tonmoy et al. 2024; Liu et al. 2021; Huang et al. 2023; Ji et al. 2023; Rawte et al. 2023), and training data contamination (Brown, et al. 2020; Shi et al. 2023; Zhang et al. 2024) becomes increasingly important to their reliable deployment and responsible use. Specifically, the tasks of Hallucination and Data Contamination Detection (resp. HD, DCD) relate to determining whether an LLM is fabricating information or is providing an incorrect answer to a user question, or whether it has been exposed to specific training data, such as copyrighted material.

Prominent methods to tackle HD include probing techniques which, although effective, require restrictive white-box access to model internals (Belinkov 2022; Orgad et al. 2024; Hewitt and Manning 2019; Hewitt and Liang 2019; Rateike et al. 2023), such as its hidden states. On both HD and DCD, gray-box methods relax these assumptions by operating *only* on LLM outputs, thus finding application to a broader set of models. These approaches (Guerreiro, Voita, and Martins 2022; Kadavath et al. 2022; Varshney et al. 2023; Huang et al. 2023) typically extract simple features on the sequence of token probabilities, a vector we term Actual Token Probabilities (ATP). However, these methods, often based on heuristics, *overlook* the information contained in the complete next-token probability distributions generated over the token vocabulary at each generation step – we term this matrix the Token Distribution Sequence (TDS), see Figure 1. Importantly, this limitation can mask distinctive patterns in the model's text generation process, including its confidence or uncertainty, known to correlate with its correctness (Kuhn, Gal, and Farquhar 2023; Farquhar et al. 2024). This aspect is evident even at the level of a single generation step. Consider, e.g., an LLM generating a token with probability 0.5 in two scenarios: in one case, the remaining next-token probability mass is concentrated on a single alternative $(0.5, 0.5, 0, ..., 0)$, while in the other it is spread across many tokens: $(0.5, 0.01, ..., 0.01)$. See Figure 12 in Appendix F for an illustration of a similar case. Yet ATP-based approaches would treat them identically. Similarly, an ATP value of 0.1 at a certain time step could indicate either high uncertainty (if it is the highest probability in a diffused distribution) or strong evidence against the token (if it is a low-ranking probability in a peaked distribution). A recent promising approach (Zhang et al. 2024) used some TDS

[*]These authors contributed equally.

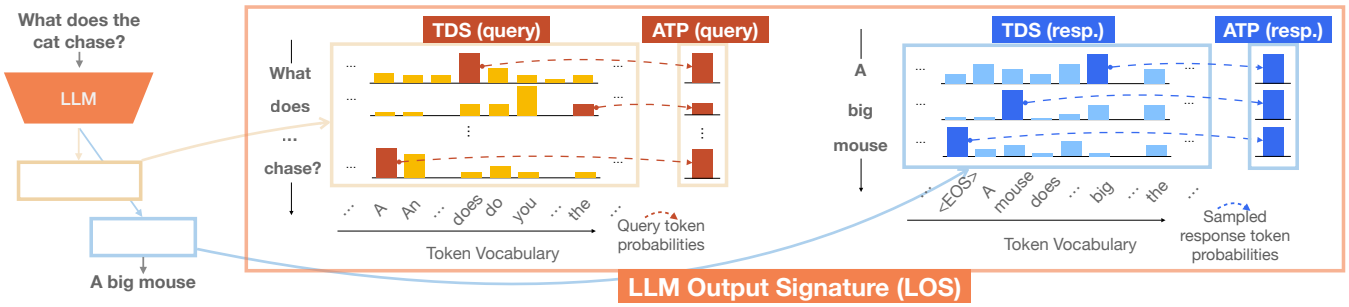

Figure 1: Left: The LLM processes the input "What does the cat chase?" and generates the output "A big mouse". Right: The corresponding query/response Token Distribution Sequences (TDS) and Actual Token Probabilities (ATP), together constituting the LLM Output Signature (LOS). We propose to detect instances of hallucinations and data contamination by learning directly over this unified data representation, beyond task specific heuristics operating on partial information thereof.

information using heuristics, but a *principled framework* to utilize this data source is still lacking.

Our approach. We argue that a successful gray-box detection approach should leverage both ATP and TDS, together forming what we term the LLM Output Signature (LOS) (Figure 1) – the complete observable representation of an LLM in the gray-box setup. Instead of relying on heuristics, we treat LOS as a sequential, high-dimensional and structured data modality on which we apply principled deep learning techniques. We propose LOS-NET, an efficient attention-based model[1] operating on an effective encoding of ATP, TDS, and their interactions. We prove that LOS-NET can approximate a broad class of functions applied to the LOS, subsuming many recent approaches (Guerreiro, Voita, and Martins 2022; Kadavath et al. 2022; Varshney et al. 2023; Huang et al. 2023; Shi et al. 2023; Zhang et al. 2024). Our comprehensive empirical study on DCD and HD demonstrates a substantial performance gap between using the complete LOS and relying solely on the ATP. Notably, LOS-NET improves over all considered baselines across both tasks, often by a significant margin. Crucially, our architecture is extremely efficient, with detection times of $\sim 10^{-5}$s per instance. This makes it a compelling approach for applications such as on-line error detection for guided-generation, as opposed to previously proposed popular methods based on multiple LLM prompting or generations, such as Semantic Entropy (SE) and P(True) (Kuhn, Gal, and Farquhar 2023; Kadavath et al. 2022). LOS-NET also exhibits promising dataset-level transfer and strong cross-model generalization, the latter suggesting its viable application to real-world tasks such as copyright-infringement detection over closed-source LLMs (see, e.g., our results on the BookMIA benchmark (Shi et al. 2023) in Section 5.2). Last, we show LOS-NET retains strong performance also when processing a small subset of the TDS, expressed in terms of the number of top-scoring output probabilities at each generation step. This extends its successful application to LLMs with restricted API access, such as GPT models (OpenAI 2024).

[1]Our model features around 1M parameters.

Contributions. (1) We introduce LOS as a suitable "data type" for the detection of hallucinations and data contamination, and develop LOS-NET, an effective and efficient learning framework for that. (2) We show this framework unifies and generalizes previous approaches, and demonstrate it achieves superior performance across models, datasets, and tasks. (3) We find that LOS-NET exhibits strong empirical evidence for cross-model generalization and promising cross-dataset transfer abilities.

2 Related Work

We review background and related work on DCD and HD, focusing on studies leveraging logits or output probabilities. Given the breadth of research, we highlight the most relevant works for our setup and refer interested readers toAppendix C for further details on these tasks.

Data Contamination Detection (DCD). This task consists in identifying text passages an LLM has likely seen during training, or memorized. This is crucial for ensuring fair benchmarking of LLMs, guiding dataset curation, and auditing potential copyright infringement. Early methods leveraged model loss (Yeom et al. 2018; Carlini et al. 2019), assuming that models overfit their training data. Later refinements introduced *reference models* – independent LLMs trained on disjoint datasets from a similar distribution – comparing their scores with the target model (Carlini et al. 2021; Carlini et al. 2022). However, this approach requires access to a well-matched reference model with similar architecture, which is often impractical in real-world settings. Recently, (Shi et al. 2023) introduced Min-K%, which flags an input as contaminated if the log probability of its bottom K tokens exceeds a predefined threshold. Building on this approach, (Zhang et al. 2024) proposed Min-K%++, which refines contamination detection by calibrating the next-token log-likelihood using the mean and standard deviation of log-likelihoods across all candidate tokens in the vocabulary.

Hallucination Detection (HD). This task has been studied to enable selective intervention, allowing LLMs to prevent fabricated outputs only when necessary (Snyder, Moisescu, and Zafar 2024; Yin, Srinivasa, and Chang 2024; Valentin et al. 2024). Recently, (Orgad et al. 2024; Azaria and Mitchell

2023) showed that training a classifier on top of LLMs' hidden states is highly effective for hallucination detection. However, these methods operate under the white-box assumption, requiring full access to the model's internal states. In contrast, our paper explores a more constrained (gray-box) setting, of particular interest especially when targeting closed-source LLMs with restricted API access.

Output probability-Based Analysis. Previous works showed that using log probabilities or raw logits as decision thresholds can be effective for various tasks, including HD in LLMs (Guerreiro, Voita, and Martins 2022; Varshney et al. 2023), correctness self-evaluation (Kadavath et al. 2022), uncertainty estimation (Huang et al. 2023), and zero shot learning (Atzmon and Chechik 2019). However, these approaches often rely on naive handcrafted thresholding. Other works (Verma et al. 2024; Mosca et al. 2022) rely on linear classifiers over features extracted from LLM outputs aiming at tackling adjacent tasks, such as detecting machine-generated text (Wu et al. 2023) but overlook the full TDS, limiting contextual understanding.

3 Learning on LLM Output Signatures

In this section, we define the LLM Output Signature (LOS) and introduce LOS-NET, a novel architecture specifically designed to process LOS.

3.1 LLM Output Signatures (LOS)

Let f denote a pretrained LLM, and $\vec{s}$ a text input to f consisting of n tokens. When queried with $\vec{s}$, f produces outputs $\mathbf{X}_s = f(\vec{s})$, i.e., a matrix in $\mathbb{R}^{n \times V}$ of next-token probabilities for each token in $\vec{s}$, where V is the size of the token vocabulary. We define $\vec{g}$ to be the LLM response to $\vec{s}$, consisting of m tokens generated using f's outputs in $\mathbf{X}_g \in \mathbb{R}^{m \times V}$ (and $\mathbf{X}_s$). We refer to $\mathbf{X}_s$ or $\mathbf{X}_g$ as *Token Distribution Sequences* (TDS). See Figure 11, Appendix E. We also define $\mathbf{p}_s \in \mathbb{R}^n, \mathbf{p}_g \in \mathbb{R}^m$, vectors which holds the probabilities associated with the actual tokens appearing in $\vec{s}, \vec{g}$ respectively. We denote these as the *Actual Token Probabilities* (ATP). Specifically, $(\mathbf{p}_s)_i \coloneqq \mathbf{X}_{i,v}$ where $v \in \{1, \ldots, V\}$ is the token used in the $i+1$ place in the sequence $\vec{s}$ and similarly for $\vec{g}$ (see Figure 1). We call the pairs $(\mathbf{X}_s, \mathbf{p}_s)$ or $(\mathbf{X}_g, \mathbf{p}_g)$ the *LLM Output Signature* (LOS). For DCD, we analyze input sequences using $(\mathbf{X}_s, \mathbf{p}_s)$ as our interest lies in how the model processes the input text $\vec{s}$. For HD, we use $(\mathbf{X}_g, \mathbf{p}_g)$ as we need to make predictions on the model's response. We may use $(\mathbf{X}, \mathbf{p})$ if the distinction between the tasks is irrelevant, and use N as the sequence length.

Problem Statement. LOS elements, along with their associated annotations depending on the task of interest, can be gathered into datasets $D = \{\big((\mathbf{X}, \mathbf{p})_i, y_i\big)\}_{i=1}^{\ell}$ where supervised learning problems can be instantiated. Our goal in this paper is to propose a neural architecture that can effectively utilize the complete LOS to solve tasks such as DCD, HD, or any other classification problem thereon.

3.2 LOS-NET

Learning from LOS data objects presents inherent challenges, particularly related to their encoding. Next, we detail these challenges and introduce our LOS-NETapproach, illustrated in full in Appendix D and Figure 10. What follows is a detailed explanation of each of its components.

Preprocessing the token distribution sequences. Utilizing $\mathbf{X}$ may pose significant challenges due to three key factors. *(1) Complexity:* The vocabulary tensor can be extremely large in real-world scenarios. For instance, Liang et al. (2023) (XLM-V) reported a vocabulary size of 1M tokens, which, for a small batch of documents and popular context sizes, would already entail processing a tensor of tens (or hundreds) of GBs. *(2) Transferability:* Vocabulary size and order may significantly vary between LLMs, something which can complicate transfer learning – e.g., training on one LLM and applying on another with a different vocabulary size; *(3) Limited Access:* As already mentioned, in certain LLMs, such as those released by OpenAI, the output tensor $\mathbf{X}$ is only *partially accessible*, with APIs only exposing a small number of the top (log-)probs. To tackle these challenges, we propose selecting, for each row of $\mathbf{X}$, a fixed number of elements. Specifically, we preprocess $\mathbf{X}$ by sorting each row independently and selecting the top K probabilities, as follows:

$$\mathbf{X}' = \text{row-sort}(\mathbf{X})_{:,:K}, \tag{1}$$

resulting in $\mathbf{X}' \in \mathbb{R}^{N \times K}$. This approach not only reduces computational complexity but also provides a standardized representation independent of the vocabulary size (for an appropriate choice of K). Later, in Section 5, we will show how even small values of K can capture most of the TDS probability mass and enable strong empirical performance.

Encoding the ATP. The tensor $\mathbf{X}'$ provides a comprehensive description of the LLM's output, but does not explicitly encode an important source of information: the probability $\mathbf{p}$ of the actual tokens appearing in the sequence, i.e, the ATP. While these values are technically present in the TDS (since it contains the full distribution), they are not directly distinguishable from the other token probabilities in the vocabulary. Thus, we do also include ATPs as separate inputs to our architecture and further complement these probabilities with additional information which allows us to contextualize them with respect to the whole TDS. Specifically, we argue that valuable information is encoded in the *rank*[2] (position) of the ATP within the sorted TDS. This information reveals the "gap" between the actual token and the token the model would most likely expect to find instead. We encode the rank in a way to make this feature more amenable for learning: we apply scaling to a closed interval and transform it with specific parameters, obtaining $\text{RE}(\mathbf{X}, \mathbf{p})$. More details are found in Appendix B.4.

Architecture. Given the preprocessed TDS $\mathbf{X}'$ and the rank encoding $\text{RE}(\mathbf{X}, \mathbf{p})$, we first linearly project $\mathbf{X}'$ via $\mathbf{W} \in \mathbb{R}^{K \times K'}$, concatenate it with $\text{RE}(\mathbf{X}, \mathbf{p})$, and then feed it to an encoder-only transformer module $\mathcal{T}$ with learnable positional encodings, operating in the sequence dimension

[2]The rank of the i-th token is defined as: $r_i(\mathbf{X}, \mathbf{p}) = \sum_{v=1}^{V} \mathbb{I}(\mathbf{X}_{i,v} > p_i)$, where $\mathbb{I}(\cdot)$ is the indicator function.

(Vaswani 2017):

$$h_\theta(\mathbf{X}, \mathbf{p}) = \mathcal{T}\left(\mathbf{X}'\mathbf{W} \,\middle\|\, \mathrm{RE}(\mathbf{X}, \mathbf{p})\right). \tag{2}$$

Here, θ includes all model's parameters, $\|$ denotes concatenation on the feature dimension. Finally, we pool over the [CLS] token and obtain output scores via a linear layer. The resulting model, LOS-NET, is trained with binary cross-entropy loss.

4 Generalization of Previous Approaches

As already mentioned, prior research has introduced various gray-box, methods for HD and DCD based on LLM's output probabilities (Guerreiro, Voita, and Martins 2022; Kadavath et al. 2022; Varshney et al. 2023; Huang et al. 2023). In what follows, we propose a general framework to unify these diverse techniques, and show that this can be captured by our LOS-NET method, shedding light on its flexibility.

Motivating example: Min-K% Shi et al. (2023). Min-K%, a prominent, recent method for DCD, makes predictions on an input text $\vec{s}$ based on a score R calculated as the average of the smallest $K\%$ log-probs: $R(\vec{s}) = \frac{1}{|M|}\sum_{i\in M}\log(p_i)$, with $M = \{i \mid p_i < \mathrm{perc}(\mathbf{p}, K)\}$ being the set of token indices whose probabilities are in the first K-th percentile of $\mathbf{p}$. We note that it is instructive to rewrite the scoring equation as:

$$R(\vec{s}) = \sum_{i=1}^{|\vec{s}|} \overbrace{\frac{\log(p_i)}{\lceil \frac{K}{100}\cdot|\vec{s}|\rceil}}^{\text{token-wise score}} \cdot \underbrace{\mathbb{I}\big(\overbrace{p_i}^{\text{confidence}} < \overbrace{\mathrm{perc}(\mathbf{p}, K)}^{\text{adaptive threshold}}\big)}_{\text{gating}}. \tag{3}$$

This highlights a general pattern: that of computing a global score by aggregating token-wise values meeting a (dynamic) "acceptance" condition, a form of "gating". To unify the aforementioned baselines under a common framework, we formalize this pattern via a family of functions (see next).

Gated Scoring Functions (GSFs). We define the family of *Gated Scoring Functions* (GSF) as the set of functions scoring LOSs by aggregating token-wise scores across the input sequence whenever their confidence values exceed a (possibly adaptive) threshold. GSFs are described in terms of the following components: (1) A confidence function $\kappa : \mathbb{R}^{N\times k} \times \mathbb{R}^N \to \mathbb{R}^N$ that assigns confidence values to each token in the sequence; (2) A threshold function $T : \mathbb{R}^{N\times k} \times \mathbb{R}^N \to \mathbb{R}$ that determines an acceptance criterion; and (3) A weight function $g : \mathbb{R}^{N\times k} \times \mathbb{R}^N \to \mathbb{R}^N$ that assigns importance scores to tokens. Given a LOS $(\mathbf{X}, \mathbf{p})$, a GSF computes a global score $R(\mathbf{X}, \mathbf{p})$ as follows:

$$F(\mathbf{X}, \mathbf{p})_i = \begin{cases} g(\mathbf{X}', \mathbf{p})_i, & \text{if } \kappa(\mathbf{X}', \mathbf{p})_i \geq T(\mathbf{X}', \mathbf{p}), \\ 0, & \text{otherwise,} \end{cases},$$

$$R(\mathbf{X}, \mathbf{p}) = \sum_{i=1}^{N} F(\mathbf{X}, \mathbf{p})_i, \tag{4}$$

where $\mathbf{X}'$ is the sorted version of $\mathbf{X}$, as per Equation (1). The family of GSF is flexible enough to capture previously proposed gray-box methods, as we show in the following:

Proposition 4.1 (GSFs capture known baselines)**.** *Let $\mathcal{B}$ be the set of scoring functions implemented by the Min/Max/Mean aggregated probability methods (Guerreiro, Voita, and Martins 2022; Kadavath et al. 2022; Varshney et al. 2023; Huang et al. 2023) for HD, as well as Loss (Yeom et al. 2018), the MinK% (Shi et al. 2023) and MinK%++ (Zhang et al. 2024) methods for DCD. For any scoring function $f \in \mathcal{B}$, there exists a choice of functions κ, T, g such that the GSF R in Equation* (4)*, implements f.*

It is easy to see, e.g., how MinK% is implemented as a GSF[3]. Refer to Appendix A for more details on how other baselines are implemented.

LOS-NET can approximate GSFs and implement known baselines. As the following result shows, our LOS-NET can, in fact, approximate virtually all GSFs of interest; intuitively, there exist sets of parameters such that it evaluates "arbitrarily close" to the target GSFs.

Proposition 4.2 (LOS-NET can approximate Equation (4))**.** *Assume maximal possible vocabulary size V_{max} and context size N_{max}. Let $\mathcal{X} \times \mathcal{M} \subseteq \mathbb{R}^{N_{max}\times V_{max}} \times \mathbb{R}^{N_{max}}$ represent a compact subset in the LOS. For any measurable $\kappa : \mathcal{X} \times \mathcal{M} \to \mathbb{R}^{N_{max}}$, measurable $T : \mathcal{X} \times \mathcal{M} \to \mathbb{R}$, measurable and integrable weight function $g : \mathcal{X} \times \mathcal{M} \to \mathbb{R}^{N_{max}}$, and for any $\epsilon > 0$, there exists a set of parameters θ such that our model $h_\theta : \mathcal{X} \times \mathcal{M} \to \mathbb{R}$ satisfies $\|h_\theta - R\|_{L_1} < \epsilon$ where $\|\cdot\|_{L_1}$ denotes the L_1 norm.*

To prove this result, we build on existing universality results on approximating continuous functions with Transformers (Yun et al. 2019), showing that our (generally non-continuous) target functions can be approximated by continuous functions. Importantly, Proposition 4.2 implies that, as long as the LOS space of interest lies within a compact domain[4], our model can approximate the general GSF in Equation (4) of LOSs for any LLM under mild conditions on κ, T, and g. Note that Proposition 4.2 cannot be generally extended to L_∞ due to the discontinuity of GSFs. The practical relevance of Proposition 4.2, is underscored by the following:

Corollary 4.3 (Approximation of Baselines by LOS-NET)**.** *Our architecture, as defined in Equation* (2)*, can arbitrarily well approximate, in the L_1 sense, any of the baseline methods in $\mathcal{B}$ when operating on context and token-vocabulary of, resp., maximal sizes N_{max} and V_{max}.*

The above states that well-established, successful baselines (see class $\mathcal{B}$ in Proposition 4.1) can be approximated by LOS-NET. All proofs are enclosed in Appendix A.

5 Experiments

We assess various aspects of learning with LOS via the following questions: (1) Is learning on LOS an effective approach for DCD and HD? Does it outperform baselines?

[3]For a sequence length of N, it suffices to choose: $T(\mathbf{X}', \mathbf{p}) = -\mathrm{perc}(\mathbf{p}, K) = -\big(\mathrm{sort}(\mathbf{p})_{\lceil \frac{K}{100}\cdot N\rceil}\big)$, $\kappa(\mathbf{X}', \mathbf{p}) = -\mathbf{p}$, $g(\mathbf{X}', \mathbf{p}) = \frac{\log \mathbf{p}}{\lceil \frac{K}{100}\cdot N\rceil}$.

[4]This is inherently satisfied when using probabilities; or via clamping in the case of logits or log-probs.

And how important is $\mathbf{X}$, i.e., the TDS, in the pair $(\mathbf{X}, \mathbf{p})$, often overlooked? (Sections 5.1 and 5.2); (2) Does our model exhibit transfer capabilities across LLMs and datasets? (Section 5.3); (3) What is the practical runtime of our approach and how robust is it w.r.t. K (Appendix B.10 and section 5.4)? In the following, we present our main results, and refer to Appendix B for additional results and details.

General setup. Our experiments focus on the two tasks of DCD and HD; in all results presented next, hyperparameter K is always set to $1,000$, while its impact is discussed in Section 5.4. Aligning with prior work, we use datasets and LLMs from (Shi et al. 2023; Zhang et al. 2024) for DCD and (Orgad et al. 2024) for HD, where we also experiment with an additional LLM (Qwen-2.5-7b-Instruct (Yang et al. 2024)). Further details are in subsequent sections. As performance metric, we use the area under the ROC curve (AUC), a standard metric in this domain (Orgad et al. 2024; Shi et al. 2023; Zhang et al. 2024). We run each experiment with three different random seeds (when applicable) and report the mean along with the standard deviation of the results. All LOS-NET experiments were conducted using PyTorch (Paszke et al. 2019) on a single NVIDIA L-40 GPU.

Newly introduced learning-based baselines. In addition to task-specific baselines, we also introduce two novel learning-based baselines to appreciate the contribution of the TDS: ATP+R-MLP, ATP+R-TRANSF.. Specifically, we ablate information about the TDS and only process the ATP and rank information with, resp., an MLP or Transformer backbone. Formal definitions are in Appendix B.4.

5.1 Hallucination Detection

Datasets and LLMs. We adopt datasets from Orgad et al. (2024), following the same setup: the objective is to predict whether an LLM-generated response to a given input prompt is correct or not. We choose three datasets spanning various domains and tasks: HotpotQA without context (Yang et al. 2018), IMDB sentiment analysis (Maas et al. 2011), roles in Movies (Orgad et al. 2024). Details regarding the annotation process, splits and dataset sizes are in Appendix B.5. As the target LLMs, coherently with Orgad et al. (2024), we use Mistral-7b-instruct-v0.2 (Jiang et al. 2023) (Mis-7b) and LlaMa3-8b-instruct (Touvron et al. 2023) (L-3-8b), and further experiment with Qwen-2.5-7b-Instruct (Yang et al. 2024) (Q-2.5-7b).

HD Baselines. Below we enlist our baselines, please refer to the extended version of the paper for more details: (1) Aggregated probabilities/logits (Guerreiro, Voita, and Martins 2022; Kadavath et al. 2022; Varshney et al. 2023; Huang et al. 2023); (2) P(True) – Kadavath et al. (2022); (3) Semantic Entropy (Farquhar et al. 2024; Kuhn, Gal, and Farquhar 2023); (4) Activation Probes (Orgad et al. 2024; Azaria and Mitchell 2023; Belinkov 2022).

Results. Table 1 presents a comprehensive summary of results on Mis-7b and L-3-8b. These clearly demonstrate that LOS-NET outperforms all gray-box baselines across all six dataset/LLM combinations, often by a significant margin.

Method	HotpotQA	IMDB	Movies
	Mistral-7b-instruct		
Logits-mean	61.00 ± 0.20	57.00 ± 0.60	63.00 ± 0.50
Logits-min	61.00 ± 0.30	52.00 ± 0.70	66.00 ± 0.80
Logits-max	53.00 ± 0.80	47.00 ± 0.40	54.00 ± 0.40
Probas-mean	63.00 ± 0.30	54.00 ± 0.80	61.00 ± 0.20
Probas-min	58.00 ± 0.30	51.00 ± 1.00	60.00 ± 0.80
Probas-max	50.00 ± 0.50	48.00 ± 0.40	51.00 ± 0.50
P(True)	54.00 ± 0.60	62.00 ± 0.90	62.00 ± 0.50
Semantic Entropy	67.66 ± 0.55	62.44 ± 0.81	<u>70.24</u> ± 0.68
ATP+R-MLP	68.92 ± 0.24	90.70 ± 0.50	66.04 ± 0.13
ATP+R-TRANSF.	<u>69.70</u> ± 0.39	<u>89.64</u> ± 1.08	67.92 ± 0.98
LOS-NET	**72.92** ± 0.45	**94.73** ± 0.58	**72.20** ± 0.66
Act. Probe †	73.00 ± 0.60	92.00 ± 1.00	72.00 ± 0.50
	Llama3-8b-instruct		
Logits-mean	65.00 ± 0.20	59.00 ± 1.70	75.00 ± 0.50
Logits-min	67.00 ± 0.80	55.00 ± 1.60	71.00 ± 0.50
Logits-max	59.00 ± 0.50	51.00 ± 0.90	67.00 ± 0.30
Probas-mean	61.00 ± 0.20	73.00 ± 1.50	73.00 ± 0.60
Probas-min	60.00 ± 0.40	57.00 ± 1.60	65.00 ± 0.40
Probas-max	56.00 ± 0.50	49.00 ± 0.80	64.00 ± 0.60
P(True)	55.00 ± 0.50	60.00 ± 0.60	66.00 ± 0.40
Semantic Entropy	65.58 ± 0.53	74.96 ± 1.00	72.27 ± 0.65
ATP+R-MLP	64.50 ± 0.75	<u>88.68</u> ± 0.30	73.25 ± 0.15
ATP+R-TRANSF.	<u>66.72</u> ± 0.39	85.46 ± 1.14	<u>75.89</u> ± 1.07
LOS-NET	**72.60** ± 0.34	**90.57** ± 0.28	**77.43** ± 0.66
Act. Probe †	77.00 ± 0.50	81.00 ± 1.40	78.00 ± 0.40

Table 1: Test AUCs for HD over Mis-7b and L-3-8b (**bold**: best method, <u>underlined</u>: second best). orange : baselines requiring additional prompting/generations. † Activation Probes , included as reference: *incomparable as they access model internals*.

These also include P(True) and Semantic Entropy, which use auxiliary prompts or generations. We highlight how, on the IMDB dataset, LOS-NET achieves an AUC improvement of around 31 units over the best of these baselines for Mis-7b and 17 over the best baseline for L-3-8b. Intriguingly, we note how LOS-NET outperforms even white-box Activation Probes in 2 out of 6 combinations, while performing similarly in 3 of them. Our results also indicate that ATP learning-based baselines consistently underperform compared to LOS-NET, underscoring the critical role of the TDS, $\mathbf{X}$. Our ATP-based learnable baselines still outperform non-learnable probability-based methods in most cases, suggesting that a learning approach relying exclusively on ATP can still be a viable solution in certain scenarios. Results on Q-2.5-7b are consistent with the above findings, and are deferred to Appendix B.8.

5.2 Data Contamination Detection

DCD is often framed as a Membership Inference Attack (MIA) (Shokri et al. 2017; Mattern et al. 2023; Shi et al. 2023). A DC dataset $D = \{q_i, y_i\}_{i=1}^{\ell}$ contains ℓ text sam-

ples, where q_i represents the text and y_i, the target, indicates whether it was part of the training data or not.

Datasets and LLMs. We use three datasets to assess DCD, specifically: WikiMIA-32 and WikiMIA-64 (Shi et al. 2023) (excerpts from Wikipedia articles), as well as BookMIA (Shi et al. 2023) (excerpts from books). Henceforth, due to space limitations, we will only discuss details and results related to the latter, while referring readers to Appendix B.9 for the former. In BookMIA, positive members correspond to books known to be well memorized by certain OpenAI models (Chang et al. 2023), or otherwise known to (partly) be in the pretraining corpus of other open-source LLMs (Antebi et al. 2025). Non-members include excerpts from books released after 2023, necessarily absent from the pretraining corpus of the last ones. This dataset allows us to test LOS-NET in a realistic scenario akin to copyright-infringement detection. In particular, contrary to previous works, we propose a novel split that ensures all excerpts from the same book always appear either in the training or test split (and never in both). Details are enclosed in Appendix B.5. We attack LLMs considered in (Antebi et al. 2025): LlaMa-13b/30b (Touvron et al. 2023) (L-13b/30b), Pythia-6.9b/12b(Biderman et al. 2023) (P-6.9b/12b).

DCD Baselines. : (1) the Loss approach (Yeom et al. 2018); (2) the Reference-based perplexity calibration (Ref) (Carlini et al. 2021); (3) Zlib and (4) Lowercase (Carlini et al. 2021);(5) Min-K% (Shi et al. 2023) and (6) Min-K%++ (Zhang et al. 2024). Note: (1, 5, 6) are reference-free; (2–4) are reference-based, i.e., they compare scores of the target LLM with those of a reference[5]. Please refer to the extended version of the paper for more details.

Results. Refer to Table 2 for results on BookMIA. LOS-NET attains exceptional results, largely surpassing other reference-free approaches. Among these last ones, ours is the only method that can match or outperform even the reference-LLM-based baselines. Importantly, (even partial) access to the TDS reveals crucial to obtain such strong reference-free performance: our ATP-based learnable methods – which only process features for the actual sequence tokens – incur indeed significant performance drops.

Method / LLM	P-6.9b	P-12b	L-13b	L-30b
Loss	67.40	76.27	76.23	89.18
MinK	68.78	77.32	75.36	89.61
MinK++	66.73	71.76	72.87	80.60
Zlib	50.01	60.84	61.94	80.83
Lowercase	74.97	81.64	67.80	82.18
Ref	89.52	**91.93**	84.58	94.93
ATP+R-MLP	56.31 ± 1.48	57.18 ± 1.06	66.60 ± 1.05	83.89 ± 0.41
ATP+R-TRANSF.	79.59 ± 0.61	74.77 ± 0.57	74.65 ± 0.79	87.62 ± 0.68
LOS-NET	**90.71** ± 0.90	89.43 ± 0.59	**91.02** ± 0.15	**95.60** ± 0.41

Table 2: BookMIA. 'P': Pythia, 'L': LlaMa-1 (**bold**: best, underlined: second best, pink : reference-based).

As for WikiMIA, while full results are enclosed in Appendix B.9 (Table 6), we point out how LOS-NET consistently surpasses all baselines across all combinations of LLMs and datasets. We also report the second-best method is MinK%++, followed by MinK%, consistent with the findings in (Zhang et al. 2024).

[5]For example for Pythia-12b, a valid reference LLM would be the smaller Pythia-70M.

5.3 Generalization to Other LLMs and Datasets

Zero-shot Cross-LLM Generalization in DCD. We assess our model's ability to detect DC in target LLMs that were unseen during training. Using the BookMIA benchmark and the setup described in Section 5.2, we evaluate our model directly across different LLMs *without any fine-tuning*. This setup is particularly relevant in DCD scenarios where contamination information is not yet available for newly released LLMs. The results are presented in the heatmap shown in Figure 2. We observe strong transferability: in 10/12 cases, our model achieves the best performance among reference-free approaches, highlighted in bold in Figure 2. Interestingly, in 3/12 cases, LOS-NET (which is reference-free) even surpasses reference-based baselines, as indicated via a superscript of *. We also observe particularly strong transfer across differently sized LLM architectures within the same family and highlight the surprising positive transfer from the largest LlaMa to Pythia models.

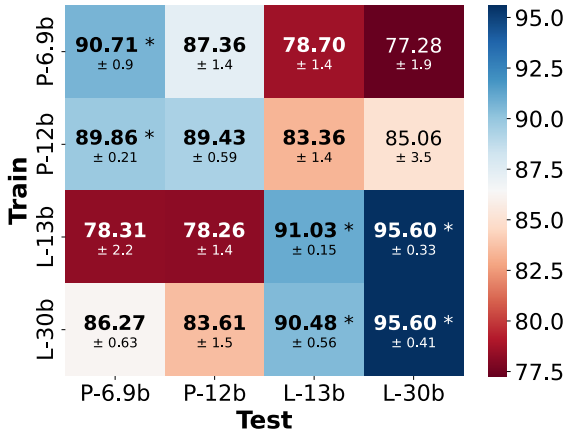

Figure 2: BookMIA zero-shot AUC – **bold**, *: outperforms, resp., ref-free and ref-based.

Transfer Learning across LLMs and Datasets for HD. Although LOS-NET delivered non-trivial generalization, its zero-shot application on HD was not sufficient to surpass the simpler probability-based techniques. This led us to investigate LOS-NET capabilities in a transfer learning setting. Specifically, we fix an LLM/dataset combination and fine-tune the corresponding pretrained LOS-NET either on the remaining LLMs for the same dataset, or the remaining datasets for the same LLM. All Test AUCs of our fine-tuned LOS-NET's are in Figures 4 and 5, Appendix B.6, while we report here two representative plots (see Figure 3). On these heatmaps, superscript '*' indicates the fine-tuned LOS-NET is better than a counterpart trained from scratch in the same

setting – testing for successful transfer; **bold** indicates it outperforms the best non-learnable probability-based method.

Discussion. First, LOS-NET exhibits solid transferability in both scenarios. The finetuned models consistently outperform their counterparts trained from scratch: 16/18 cases in both the cross-LLM (Figure 4, Appendix B.6) and cross-dataset setups (Figure 5, Appendix B.6) – see '*' on the off-diagonal entries. This highlights a generally positive transfer of LOS-NET's learned representations across datasets and LLMs, and underscores the suitability of LOS as a data type in capturing generalizable patterns for HD. Second, from a practical perspective, we find that LOS-NET outperforms the best probability-based baseline in 15/18 cases for both the cross-LLM (Figure 4) and cross-dataset (Figure 5) scenarios – see **bold** on the off-diagonal entries. Focusing on the IMDB dataset, when training on L-3-8b and testing on Mis-7b (Figure 3 (bottom)), our model substantially gains around 27 AUC units over the best probability-based baseline. This result underscores the possibility of transferring across LLMs. A similar trend is observed in the cross-dataset setup (Figure 3 (top)): on Mis-7b, when training on HotpotQA or Movies and testing on IMDB, our model achieves a notable improvement of around 30 AUC units compared to the best baseline).

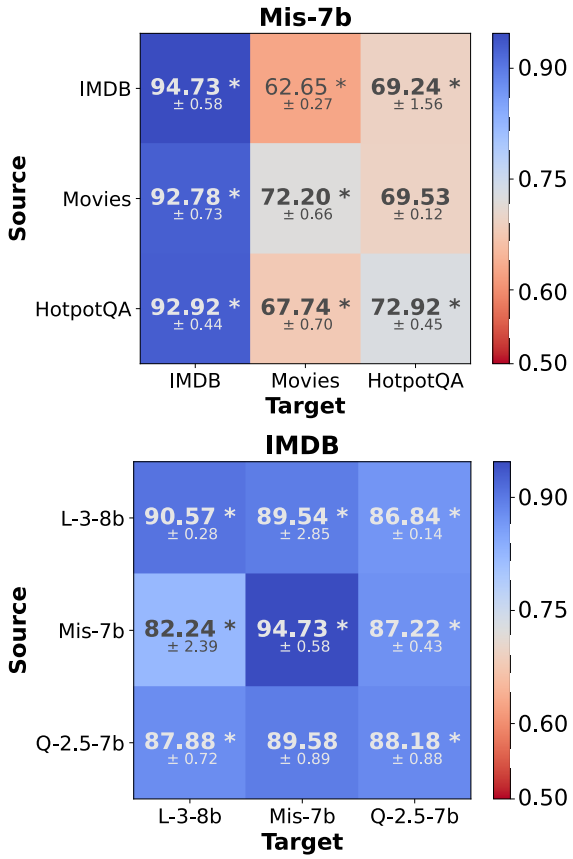

Figure 3: Transfer Test AUC to varying datasets (top, Mis-7b) and LLMs (bottom, IMDB fixed).

5.4 The value of K and restricted TDS access

We conclude this section by presenting results on the impact of the parameter K, as defined in Equation (1). In particular, we slide K in $\{10, 50, 100, 500, 1000\}$ and discuss here results for the Mis-7B LLM on the HotpotQA dataset, see Table 3. For each of the above values, we measure two quantities: the average probability mass captured, i.e., $\sum_{n=1}^{N}(\sum_{v=1}^{V}(\mathbf{X}'_{n,v}))/N$, and the corresponding performance of LOS-NET. We observe that the former exceeds 0.99 for all considered K's, indicating that even values as small as $K = 10$ are often sufficient to convey most of the information in the full TDS. In terms of performance, Test AUC tends to improve as K values increase, though with diminishing returns beyond $K \geq 100$. As expected, the value $K = 1000$ tends to deliver the best performance overall, this confirmed by results on the other LLM-dataset combinations reported in Appendix B.7. Given its extremely contained run-times (see above), this value appears to hit a sweet-spot optimizing performance and complexity. Most notably, however, even with $K = 10$, LOS-NET outperforms all baselines and matches the white-box *Activation Probe*, highlighting the practical effectiveness of LOS-NET even in API-limited settings such as in GPT models – at the time of writing, exposing only the top $K = 20$ output logits.

K	APM (%)	Test AUC
10	99.49	71.82 ± 0.15
50	99.80	71.87 ± 0.24
100	99.85	72.34 ± 0.20
500	99.99	72.67 ± 0.32
1000	99.99	**72.92** ± 0.45

Table 3: Ablation study on K. Average Probability Mass (APM) and AUC for varying K on Mis-7B - HotpotQA.

6 Conclusions

We proposed LOS-NET, an efficient method to detect data contamination and hallucinations in LLMs by leveraging their output signatures (LOS), defined as the union of Token Distribution Sequences (TDS) and Actual Token Probabilities (ATP). LOS-NET consists of a lightweight attention-based model operating on an effective encoding of the LOS. We proved it unifies and extends existing gray-box methods under a general framework, and experimentally showed it outperforms state-of-the-art gray-box methods across datasets and LLMs. It also exhibited promising generalization and transfer capabilities of LOS-NET, both across datasets and across LLMs. Our framework could be applied to other tasks, such as detecting LLM-generated content. Additional sources of information can also be incorporated, e.g., in the absence of latency constraints, it can be interesting to include "exact-token" flags as proposed by (Orgad et al. 2024). Last, the LOS can be extended to account for multiple prompting (Kuhn, Gal, and Farquhar 2023).

Acknowledgments

The authors are grateful to Beatrice Bevilacqua for insightful discussions. G.B. is supported by the Jacobs Qualcomm PhD Fellowship. F.F. conducted this work supported by an Aly Kaufman and an Andrew and Erna Finci Viterbi Post-Doctoral Fellowship. Y.G. is supported by the UKRI Engineering and Physical Sciences Research Council (EPSRC) CDT in Autonomous and Intelligent Machines and Systems (grant reference EP/S024050/1). H.M. is a Robert J. Shillman Fellow and is supported by the Israel Science Foundation through a personal grant (ISF 264/23) and an equipment grant (ISF 532/23). D.L. is funded by an NSF Graduate Fellowship. Research was also supported by the Israeli Ministry of Science, Israel-Singapore binational grant 207606. F.F. is extremely grateful to the members of the "Eva Project", whose support he immensely appreciates.

References

Antebi et al. 2025. Tag&Tab: Pretraining Data Detection in Large Language Models Using Keyword-Based Membership Inference Attack. *arXiv preprint arXiv:2501.08454*.

Atzmon, Y.; and Chechik, G. 2019. Adaptive confidence smoothing for generalized zero-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11671–11680.

Azaria, A.; and Mitchell, T. 2023. The Internal State of an LLM Knows When It's Lying. arXiv:2304.13734.

Belinkov, Y. 2022. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*.

Biderman et al. 2023. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, 2397–2430. PMLR.

Biewald, L. 2020. Experiment Tracking with Weights and Biases.

Brown, et al. 2020. Language Models are Few-Shot Learners. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M.; and Lin, H., eds., *Advances in Neural Information Processing Systems*, volume 33, 1877–1901. Curran Associates, Inc.

Cao et al. 2022. Learning with Rejection for Abstractive Text Summarization. In Goldberg, Y.; Kozareva, Z.; and Zhang, Y., eds., *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Carlini, N.; Chien, S.; Nasr, M.; Song, S.; Terzis, A.; and Tramer, F. 2022. Membership inference attacks from first principles. In *2022 IEEE Symposium on Security and Privacy (SP)*, 1897–1914. IEEE.

Carlini, N.; Liu, C.; Erlingsson, Ú.; Kos, J.; and Song, D. 2019. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *28th USENIX security symposium (USENIX security 19)*, 267–284.

Carlini et al. 2021. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, 2633–2650.

Chang et al. 2023. Speak, Memory: An Archaeology of Books Known to ChatGPT/GPT-4. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*.

Farquhar, S.; Kossen, J.; Kuhn, L.; and Gal, Y. 2024. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017): 625–630.

Gao et al. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*.

Gu et al. 2023. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*.

Guerreiro, N. M.; Voita, E.; and Martins, A. F. 2022. Looking for a needle in a haystack: A comprehensive study of hallucinations in neural machine translation. *arXiv preprint arXiv:2208.05309*.

Hewitt, J.; and Liang, P. 2019. Designing and interpreting probes with control tasks. *arXiv preprint arXiv:1909.03368*.

Hewitt, J.; and Manning, C. D. 2019. A Structural Probe for Finding Syntax in Word Representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*.

Huang, Y.; Song, J.; Wang, Z.; Zhao, S.; Chen, H.; Juefei-Xu, F.; and Ma, L. 2023. Look before you leap: An exploratory study of uncertainty measurement for large language models. *arXiv preprint arXiv:2307.10236*.

Huang et al. 2023. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*.

Izacard et al. 2023. Atlas: Few-shot learning with retrieval augmented language models. *Journal of Machine Learning Research*, 24(251): 1–43.

Ji et al. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*.

Jiang et al. 2023. Mistral 7B. *arXiv preprint arXiv:2310.06825*.

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

Kuhn, L.; Gal, Y.; and Farquhar, S. 2023. Semantic Uncertainty: Linguistic Invariances for Uncertainty Estimation in Natural Language Generation. arXiv:2302.09664.

Lewis et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33: 9459–9474.

Li, K.; Patel, O.; Viégas, F.; Pfister, H.; and Wattenberg, M. 2024a. Inference-time intervention: Eliciting truthful answers from a language model. *Advances in Neural Information Processing Systems*, 36.

Li, Y.; Guo, Y.; Guerin, F.; and Lin, C. 2024b. An Open-Source Data Contamination Report for Large Language Models. In Al-Onaizan, Y.; Bansal, M.; and Chen, Y.-N.,

eds., *Findings of the Association for Computational Linguistics: EMNLP 2024*, 528–541. Miami, Florida, USA: Association for Computational Linguistics.

Liang, D.; Gonen, H.; Mao, Y.; Hou, R.; Goyal, N.; Ghazvininejad, M.; Zettlemoyer, L.; and Khabsa, M. 2023. Xlm-v: Overcoming the vocabulary bottleneck in multilingual masked language models. *arXiv preprint arXiv:2301.10472*.

Liu, T.; Zhang, Y.; Brockett, C.; Mao, Y.; Sui, Z.; Chen, W.; and Dolan, B. 2021. A token-level reference-free hallucination detection benchmark for free-form text generation. *arXiv preprint arXiv:2104.08704*.

Loshchilov, I. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Maas et al. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, 142–150.

Mattern, J.; Mireshghallah, F.; Jin, Z.; Schölkopf, B.; Sachan, M.; and Berg-Kirkpatrick, T. 2023. Membership inference attacks against language models via neighbourhood comparison. *arXiv preprint arXiv:2305.18462*.

Maynez et al. 2020. On Faithfulness and Factuality in Abstractive Summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 1906–1919. Online: Association for Computational Linguistics.

Mosca et al. 2022. "That Is a Suspicious Reaction!": Interpreting Logits Variation to Detect NLP Adversarial Attacks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 7806–7816. Dublin, Ireland: Association for Computational Linguistics.

OpenAI. 2024. GPT-4 Technical Report. arXiv:2303.08774.

Orgad, H.; Toker, M.; Gekhman, Z.; Reichart, R.; Szpektor, I.; Kotek, H.; and Belinkov, Y. 2024. Llms know more than they show: On the intrinsic representation of llm hallucinations. *arXiv preprint arXiv:2410.02707*.

Paszke et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Qiu, Y.; Zhao, Z.; Ziser, Y.; Korhonen, A.; Ponti, E. M.; and Cohen, S. B. 2024. Spectral Editing of Activations for Large Language Model Alignment. *arXiv preprint arXiv:2405.09719*.

Qiu et al. 2023. Detecting and Mitigating Hallucinations in Multilingual Summarisation. In Bouamor, H.; Pino, J.; and Bali, K., eds., *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Singapore: Association for Computational Linguistics.

Rateike, M.; Cintas, C.; Wamburu, J.; Akumu, T.; and Speakman, S. 2023. Weakly supervised detection of hallucinations in llm activations. *arXiv preprint arXiv:2312.02798*.

Rawte et al. 2023. The troubling emergence of hallucination in large language models–an extensive definition, quantification, and prescriptive remediations. *arXiv preprint arXiv:2310.04988*.

Shi, W.; Ajith, A.; Xia, M.; Huang, Y.; Liu, D.; Blevins, T.; Chen, D.; and Zettlemoyer, L. 2023. Detecting pretraining data from large language models. *arXiv preprint arXiv:2310.16789*.

Shokri et al. 2017. Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*. IEEE.

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

Tonmoy et al. 2024. A comprehensive survey of hallucination mitigation techniques in large language models. *arXiv preprint arXiv:2401.01313*.

Touvron et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Valentin et al. 2024. Cost-effective hallucination detection for llms. *arXiv preprint arXiv:2407.21424*.

Varshney, N.; Yao, W.; Zhang, H.; Chen, J.; and Yu, D. 2023. A stitch in time saves nine: Detecting and mitigating hallucinations of llms by validating low-confidence generation. *arXiv preprint arXiv:2307.03987*.

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

Verma et al. 2024. Ghostbuster: Detecting Text Ghostwritten by Large Language Models. 1702–1717. Mexico City, Mexico: Association for Computational Linguistics.

Wu et al. 2023. LLMDet: A third party large language models generated text detection tool. *arXiv preprint arXiv:2305.15004*.

Yang et al. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. *arXiv preprint arXiv:1809.09600*.

Yang et al. 2024. Qwen2 Technical Report. *arXiv preprint arXiv:2407.10671*.

Yeom et al. 2018. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st computer security foundations symposium (CSF)*, 268–282.

Yin, F.; Srinivasa, J.; and Chang, K.-W. 2024. Characterizing truthfulness in large language model generations with local intrinsic dimension. *arXiv preprint arXiv:2402.18048*.

Yun, C.; Bhojanapalli, S.; Rawat, A. S.; Reddi, S. J.; and Kumar, S. 2019. Are transformers universal approximators of sequence-to-sequence functions? *arXiv preprint arXiv:1912.10077*.

Zhang, J.; Sun, J.; Yeats, E.; Ouyang, Y.; Kuo, M.; Zhang, J.; Yang, H. F.; and Li, H. 2024. Min-k%++: Improved baseline for detecting pre-training data from large language models. *arXiv preprint arXiv:2404.02936*.

Zhao et al. 2024. Enhancing Contextual Understanding in Large Language Models through Contrastive Decoding. In Duh, K.; Gomez, H.; and Bethard, S., eds., *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*.