Detecting Pretraining Data from Large Language Models

Weijia Shi[1] * **Anirudh Ajith**[2]* **Mengzhou Xia**[2] **Yangsibo Huang**[2]
Daogao Liu[1] **Terra Blevins**[1] **Danqi Chen**[2] **Luke Zettlemoyer**[1]
[1]University of Washington [2]Princeton University
swj0419.github.io/detect-pretrain.github.io

Abstract

Although large language models (LLMs) are widely deployed, the data used to train them is rarely disclosed. Given the incredible scale of this data, up to trillions of tokens, it is all but certain that it includes potentially problematic text such as copyrighted materials, personally identifiable information, and test data for widely reported reference benchmarks. However, we currently have no way to know which data of these types is included or in what proportions. In this paper, we study the pretraining data detection problem: *given a piece of text and black-box access to an LLM without knowing the pretraining data, can we determine if the model was trained on the provided text?* To facilitate this study, we introduce a dynamic benchmark WIKIMIA that uses data created before and after model training to support gold truth detection. We also introduce a new detection method MIN-K% PROB based on a simple hypothesis: an unseen example is likely to contain a few outlier words with low probabilities under the LLM, while a seen example is less likely to have words with such low probabilities. MIN-K% PROB can be applied without any knowledge about the pretraining corpus or any additional training, departing from previous detection methods that require training a reference model on data that is similar to the pretraining data. Moreover, our experiments demonstrate that MIN-K% PROB achieves a 7.4% improvement on WIKIMIA over these previous methods. We apply MIN-K% PROB to three real-world scenarios, copyrighted book detection, contaminated downstream example detection and privacy auditing of machine unlearning, and find it a consistently effective solution.

1 Introduction

As the scale of language model (LM) training corpora has grown, model developers (e.g, GPT-4 (Brown et al., 2020a) and LLaMA 2 (Touvron et al., 2023b)) have become reluctant to disclose the full composition or sources of their data. This lack of transparency poses critical challenges to scientific model evaluation and ethical deployment. Critical private information may be exposed during pretraining; previous work showed that LLMs generated excerpts from copyrighted books (Chang et al., 2023) and personal emails (Mozes et al., 2023), potentially infringing upon the legal rights of original content creators and violating their privacy. Additionally, Sainz et al. (2023); Magar & Schwartz (2022); Narayanan (2023) showed that the pretraining corpus may inadvertently include benchmark evaluation data, making it difficult to assess the effectiveness of these models.

In this paper, we study the pretraining data detection problem: given a piece of text and black-box access to an LLM with no knowledge of its pretraining data, can we determine if the model was pretrained on the text? We present a benchmark, WIKIMIA, and an approach, MIN-K% PROB, for

*Equal contribution

Workshop on Regulatable Machine Learning at the 37th Conference on Neural Information Processing Systems (RegML @ NeurIPS 2023).

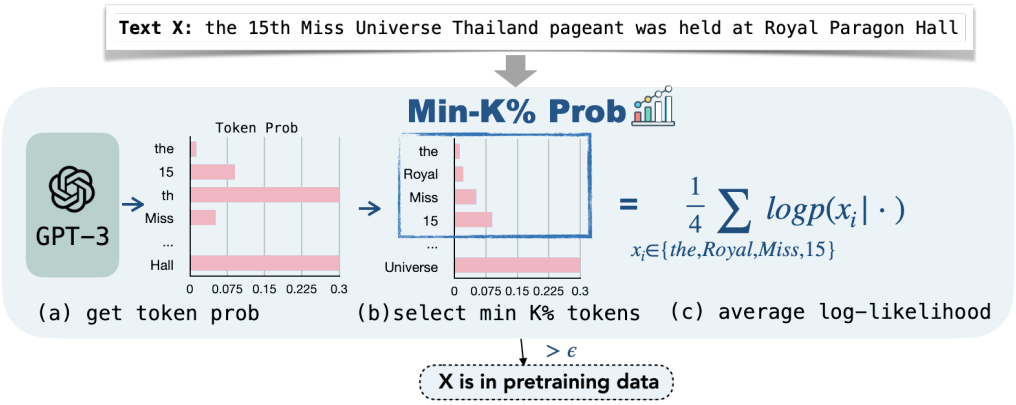


Figure 1: **Overview of MIN-K% PROB**. To determine whether a text X is in the pretraining data of a LLM such as GPT, MIN-K% PROB first gets the probability for each token in X, selects the $k\%$ tokens with minimum probabilities and calculates their average log likelihood. If the average log likelihood is high, the text is likely in the pretraining data.

pretraining data detection. This problem is an instance of Membership Inference Attacks (MIAs), which was initially proposed by Shokri et al. (2016). Recent work has studied *fine-tuning* data detection (Song & Shmatikov, 2019; Shejwalkar et al., 2021; Mahloujifar et al., 2021) as an MIA problem. However, adopting these methods to detect the pertaining data of contemporary large LLMs presents two unique technical challenges: First, unlike fine-tuning which usually runs for multiple epochs, pretraining uses a much larger dataset but exposes each instance only once, significantly reducing the potential memorization required for successful MIAs (Leino & Fredrikson, 2020; Kandpal et al., 2022). Besides, previous methods often rely on one or more reference models (Carlini et al., 2022; Watson et al., 2022) trained in the same manner as the target model (e.g., on the shadow data sampled from the same underlying pretraining data distribution) to achieve precise detection. This is not possible for large language models, as the training distribution is usually not available and training would be too expensive.

Our first step towards addressing these challenges is to establish a reliable benchmark. We introduce WIKIMIA, a dynamic benchmark designed to periodically and automatically evaluate detection methods on any newly released pretrained LLMs. By leveraging the Wikipedia data timestamp and the model release date, we select old Wikipedia event data as our member data (i.e, *seen* data during pretraining) and recent Wikipedia event data (e.g., after 2023) as our non-member data (*unseen*). Our datasets thus exhibit three desirable properties: (1) **Accurate**: events that occur after LLM pretraining are guaranteed not to be present in the pretraining data. The temporal nature of events ensures that non-member data is indeed unseen and not mentioned in the pretraining data. (2) **General**: our benchmark is not confined to any specific model and can be applied to various models pretrained using Wikipedia (e.g., OPT, LLaMA, GPT-Neo) since Wikipedia is a commonly used pretraining data source. (3) **Dynamic**: we will continually update our benchmark by gathering newer non-member data (i.e., more recent events) from Wikipedia since our data construction pipeline is fully automated.

MIA methods for finetuning (Carlini et al., 2022; Watson et al., 2022) usually calibrate the target model probabilities of an example using a shadow reference model that is trained on a similar data distribution. However, these approaches are impractical for pretraining data detection due to the black-box nature of pretraining data and its high computational cost. Therefore, we propose a reference-free MIA method MIN-K% PROB. Our method is based on a simple hypothesis: an unseen example tends to contain a few outlier words with low probabilities, whereas a seen example is less likely to contain words with such low probabilities. MIN-K% PROB computes the average probabilities of outlier tokens. MIN-K% PROB can be applied without any knowledge about the pretrainig corpus or any additional training, departing from existing MIA methods, which rely on shadow reference models (Mattern et al., 2023; Carlini et al., 2021). Our experiments demonstrate that MIN-K% PROB outperforms the existing strongest baseline by 7.4% in AUC score on WIKIMIA. Further analysis suggests that the detection performance correlates positively with the *model size* and *detecting text length*.

To verify the applicability of our proposed method in real-world settings, we perform three case studies: copyrighted book detection (§5), privacy auditing of LLMs (§7) and dataset contamination

detection (§6). We find that MIN-K% PROB significantly outperforms baseline methods in both scenarios. From our experiments on copyrighted book detection, we see strong evidence that GPT-3 [2] is pretrained on copyrighted books from the Books3 dataset (Gao et al., 2020; Min et al., 2023). From our experiments on privacy auditing of machine unlearning, we use MIN-K% PROB to audit an unlearned LLM that is trained to forget copyrighted books using machine unlearning techniques (Eldan & Russinovich, 2023) and find such model could still output related copyrighted content. Furthermore, our controlled study on dataset contamination detection sheds light on the impact of pretraining design choices on detection difficulty; we find detection becomes harder when training data sizes increase, and occurrence frequency of the detecting example and learning rates decreases.

2 Pretrainining Data Detection Problem

We study pretraining data detection, the problem of detecting whether a piece of text is part of the training data. First, we formally define the problem and describe its unique challenges that are not present in prior finetuning data detection studies (§2.1). We then curate WIKIMIA, the first benchmark for evaluating methods of pretraining data detection (§2.2).

2.1 Problem Definition and Challenges

We follow the standard definition of the membership inference attack (MIA) by Shokri et al. (2016); Mattern et al. (2023). Given a language model f_θ and its associated pretraining data $\mathcal{D} = \{z_i\}_{i\in[n]}$ sampled from an underlying distribution $\mathbb{D}$, the task objective is to learn a detector h that can infer the membership of an arbitrary data point x: $h(x, f_\theta) \rightarrow \{0, 1\}$. We follow the standard setup of MIA, assuming that the detector has access to the LM only as a black box, and can compute token probabilities for any data point x.

Challenge 1: Unavailability of the pretraining data distribution. Existing state-of-art MIA methods for data detection during finetuning (Long et al., 2018; Watson et al., 2022; Mireshghallah et al., 2022a) typically use reference models g_γ to compute the background difficulty of the data point and to calibrate the output probability of the target language model : $h(x, f_\theta, g_\gamma) \rightarrow \{0, 1\}$. Such reference models usually share the same model architecture as f_θ and are trained on shadow data $D_{\text{shadow}} \subset \mathbb{D}$ (Carlini et al., 2022; Watson et al., 2022), which are sampled from the same underlying distribution $\mathbb{D}$. These approaches assume that the detector can access (1) the distribution of the target model's training data, and (2) a sufficient number of samples from $\mathbb{D}$ to train a calibration model.

However, this assumption of accessing the distribution of pretraining training data is not realistic because such information is not always available (e.g., not released by model developers (Touvron et al., 2023b; OpenAI, 2023)). Even if access were possible, pretraining a reference model on it would be extremely computationally expensive given the incredible scale of pretraining data. In summary, the pretraining data detection problem aligns with the MIA definition but includes an assumption that the detector has no access to pretraining data distribution $\mathbb{D}$.

Challenge 2: Detection difficulty. Pretraining and finetuning differ significantly in the amount of data and compute used, as well as in optimization setups like training epochs and learning rate schedules. These factors significantly impact detection difficulty. One might intuitively deduce that detection becomes harder when dataset sizes increase, and the training epochs and learning rates decrease. We briefly describe some theoretical evidence that inform these intuitions in the following and show empirical results that support these hypotheses in §6.

To illustrate, given an example $z \in D$, we denote the model output as $f_\theta(z)$ Now, take another example y sampled from $\mathbb{D} \setminus D$ (not part of the pretraining data). Determining whether an example x was part of the training set becomes challenging if the outputs $f_\theta(z)$ and $f_\theta(y)$ are similar. The degree of similarity between $f_\theta(z)$ and $f_\theta(y)$ can be quantified using the total variation distance. According to previous research (Hardt et al., 2016; Bassily et al., 2020), the bound on this total variation distance between $f_\theta(z)$ and $f_\theta(y)$ is directly proportional to the *occurrence frequency of the example x*, *learning rates*, and the *inverse of dataset size*, which implies the detection difficulty correlates with these factors as well.

[2] `text-davinci-003`.

2.2 WIKIMIA: A Dynamic Evaluation Benchmark

We construct our benchmark by using events added to Wikipedia after specific dates, treating them as non-member data since they are guaranteed not to be present in the pretraining data, which is the key idea behind our benchmark.

Data construction. We collect recent event pages from Wikipedia. **Step 1:** We set January 1, 2023 as the cutoff date, considering events occurring post-2023 as recent events (non-member data). We used the Wikipedia API to automatically retrieve articles and applied two filtering criteria: (1) the articles must belong to the event category, and (2) the page must be created post 2023. **Step 2:** For member data, we collected articles created before 2017 because many pretrained models, e.g., LLaMA, GPT-NeoX and OPT, were released after 2017 and incorporate Wikipedia dumps into their pretraining data. **Step 3:** Additionally, we filtered out Wikipedia pages lacking meaningful text, such as pages titled "Timeline of ..." or "List of ...". Given the limited number of events post-2023, we ultimately collected 394 recent events as our non-member data, and we randomly selected 394 events from pre-2016 Wikipedia pages as our member data. The data construction pipeline is automated, allowing for the curation of new non-member data for future cutoff dates.

Benchmark setting. In practice, LM users may need to detect texts that are paraphrased and edited, as well. Previous studies employing MIA have exclusively focused on detecting examples that exactly match the data used during pretraining. It remains an open question whether MIA methods can be employed to identify paraphrased examples that convey the same meaning as the original. In addition to the verbatim setting (*original*), we therefore introduce a *paraphrase setting* we leverage ChatGPT[3] to paraphrase the examples and subsequently assess if the MIA metric can effectively identify semantically equivalent examples.

Moreover, previous MIA evaluations usually mix different-length data in evaluation and report a single performance metric. However, our results reveal that data length significantly impacts the difficulty of detection. Intuitively, shorter sentences are harder to detect. Consequently, different data length buckets may lead to varying rankings of MIA methods. To investigate this further, we propose a *different-length setting*: we truncate the Wikipedia event data into different lengths—32, 64, 128, 256—and separately report the MIA methods' performance for each length segment. We describe the desirable properties in Appendix **??**.

3 MIN-K% PROB: A Simple Reference-free Pretraining Data Detection Method

We introduce a pretraining data detection method MIN-K% PROB that leverages minimum token probabilities of a text for detection. MIN-K% PROB is based on the hypothesis that a non-member example is more likely to include a few outlier words with high negative log-likelihood (or low probability), while a member example is less likely to include words with high negative log-likelihood.

Consider a sequence of tokens in a sentence, denoted as $x = x_1, x_2, ..., x_N$, the log-likelihood of a token, x_i, given its preceding tokens is calculated as $\log p(x_i|x_1, ..., x_{i-1})$. We then select the $k\%$ of tokens from x with the minimum token probability to form a set, Min-K%(x), and compute the average log-likelihood of the tokens in this set:

$$\text{MIN-K\% PROB}(x) = \frac{1}{E} \sum_{x_i \in \text{Min-K\%}(x)} \log p(x_i|x_1, ..., x_{i-1}). \quad (1)$$

where E is the size of the Min-K%(x) set. We can detect if a piece of text was included in pretraining data simply by thresholding this MIN-K% PROB result. We summarize our method in Algorithm **??** in Appendix **??**.

[3]OpenAI. `https://chat.openai.com/chat`

4 Experiments

We evaluate the performance of MIN-K% PROB and baseline detection methods against LMs such as LLaMA Touvron et al. (2023a), GPT-Neo (Black et al., 2022), and Pythia (Biderman et al., 2023) on WIKIMIA.

4.1 Datasets and Metrics

Our experiments use WIKIMIA of different lengths (32, 64, 128, 256), *original* and *paraphrase* settings. Following (Carlini et al., 2022; Mireshghallah et al., 2022a), we evaluate the effectiveness of a detection method using the True Positive Rate (TPR) and its False Positive Rate (FPR). We plot the ROC curve to measure the trade-off between the TPR and FPR and report the AUC score (the area under ROC curve) and TPR at low FPRs (TPR@5%FPR) as our metrics.

4.2 Baseline Detection Methods

We take existing reference-based and reference-free MIA methods as our baseline methods and evaluate their performance on WIKIMIA. These methods only consider sentence-level probability. Specifically, we use the *LOSS Attack* method (Yeom et al., 2018a), which predicts the membership of an example based on the loss of the target model when fed the example as input. In the context of LMs, this loss corresponds to perplexity of the example (***PPL***). Another method we consider is the neighborhood attack (Mattern et al., 2023), which leverages probability curvature to detect membership (***Neighbor***). This approach is identical to the DetectGPT (Mitchell et al., 2023) method recently proposed for classifying machine-generated vs. human-written text. Finally, we compare with membership inference methods proposed in (Carlini et al., 2021), including comparing the example perplexity to zlib compression entropy (***Zlib***), to the lowercased example perplexity (***Lowercase***) and to example perplexity under a smaller model pretrained on the same data (***Smaller Ref***). For the smaller reference model setting, we employ LLaMA-7B as the smaller model for LLaMA-65B and LLaMA-30B, GPT-Neo-125M for GPT-NeoX-20B, OPT-350M for OPT-66B and Pythia-70M for Pythia-2.8B.

4.3 Implementation and Results

Implementation details. The key hyperparameter of MIN-K% PROB is the percentage of tokens with the highest negative log-likelihood we select to form the *top-*$k\%$ set. We performed a small sweep over 10, 20, 30, 40, 50 on a held-out validation set using the LLAMA-60B model and found that $k = 20$ works best. We use this value for all experiments without further tuning. As we report the AUC score as our metric, we don't need to determine the threshold ϵ.

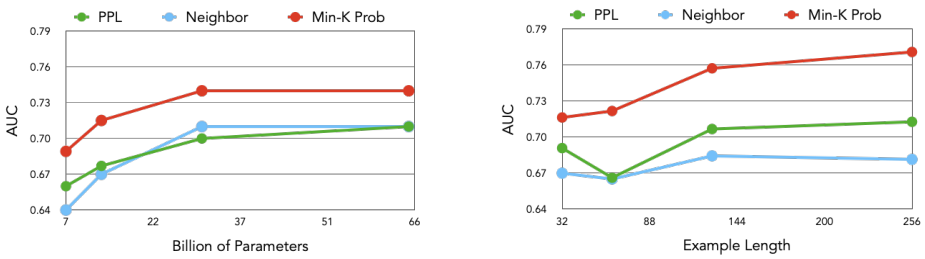
Main results. We compare MIN-K% PROB and baseline methods in Table 1. Our experiments show that MIN-K% PROB consistently outperforms all baseline methods across diverse target language models, both in original and paraphrase settings. MIN-K% PROB achieves an AUC score of 0.72 on average, marking a 7.4% improvement over the best baseline method (i.e., PPL). Among the baselines, the simple LOSS Attack (PPL) outperforms the others. This demonstrates the effectiveness and generalizability of MIN-K% PROB in detecting pretraining data from various LMs. Further results such as TPR@5%FPR can be found in Appendix A, which shows a trend similar to Table **??**.

4.4 Analysis

We further delve into the factors influencing detection difficulty, focusing on two aspects: (1) the size of the target model, and (2) the length of the text.

Model size. We evaluate the performance of reference-free methods on detecting pretraining 128-length texts from different-sized LLaMA models (7, 13, 30, 65B). Figure 2a demonstrates a noticeable trend: the AUC score of the methods rises with increasing model size. This is likely because larger models have more parameters and thus are more likely to memorize the pretraining data.

Length of text. In another experiment, we evaluate the detection method performance on examples of varying lengths in the original setting. As shown in Figure 2b, the AUC score of different methods



(a) AUC score vs. model size

(b) AUC score vs. text length

Figure 2: As model size or text length increases, detection becomes easier.

increases as text length increases, likely because longer texts contain more information memorized by the target model, making them more distinguishable from the unseen texts.

Table 1: AUC score for detecting pretraining examples from the given model on WIKIMIA for MIN-K% PROB and baselines. *Ori.* and *Para.* denote the original and paraphrase settings, respectively. **Bold** shows the best AUC within each column.

	Pythia-2.8B		**NeoX-20B**		**LLaMA-30B**		**LLaMA-65B**		**OPT-66B**		
Method	*Ori.*	*Para.*	*Ori.*	*Para.*	*Ori.*	*Para.*	*Ori.*	*Para.*	*Ori.*	*Para.*	**Avg.**
Neighbor	0.61	0.59	0.68	0.58	0.71	0.62	0.71	0.69	0.65	0.62	0.65
PPL	0.61	0.61	0.70	0.70	0.70	0.70	0.71	0.72	0.66	0.64	0.67
Zlib	0.65	0.54	0.72	0.62	0.72	0.64	0.72	0.66	0.67	0.57	0.65
Lowercase	0.59	0.60	0.68	0.67	0.59	0.54	0.63	0.60	0.59	0.58	0.61
Smaller Ref	0.60	0.58	0.68	0.65	0.72	0.64	0.74	0.70	0.67	0.64	0.66
MIN-K% PROB	**0.67**	**0.66**	**0.76**	**0.74**	**0.74**	**0.73**	**0.74**	**0.74**	**0.71**	**0.69**	**0.72**

In the following two sections, we apply MIN-K% PROB to real-world scenarios to detect copyrighted books and contaminated downstream tasks within LLMs.

5 Case Study: Detecting Copyrighted Books in Pretraining Data

MIN-K% PROB can also detect potential copyright infringement in training data, as we show in this section. Specifically, we use MIN-K% PROB to detect excerpts from copyrighted books in the Books3 subset of the Pile dataset (Gao et al., 2020) that may have been included in the GPT-3[4] training data.

5.1 Experimental Setup

Validation data to determine detection threshold. We construct a validation set using 50 books known to be memorized by ChatGPT, likely indicating their presence in its training data (Chang et al., 2023), as positive examples. For negative examples, we collected 50 new books with first editions in 2023 that could not have been in the training data. From each book, we randomly extract 100 snippets of 512 words, creating a balanced validation set of 10,000 examples. We determine the optimal classification threshold with MIN-K% PROB by maximizing detection accuracy on this set.

Test data and metrics. We randomly select 100 books from the Books3 corpus that are known to contain copyrighted contents (Min et al., 2023). From each book, we extract 100 random 512-word snippets, creating a test set of 10,000 excerpts. We apply the threshold to decide if these books snippets have been trained with GPT-3. We then report the percentage of these snippets in each book (i.e., contamination rate) that are identified as being part of the pre-training data.

[4]text-davinci-003

5.2 Results

Figure 3 shows MIN-K% PROB achieves an AUC of 0.88, outperforming baselines in detecting copyrighted books. We apply the optimal threshold of MIN-K% PROB to the test set of 10,000 snippets from 100 books from Books3. Table 2 represents the top 20 books with the highest predicted contamination rates. Figure 4 reveals nearly 90% of the books have an alarming contamination rate over 50%.

Method	Book
Neighbor	0.75
PPL	0.84
Zlib	0.81
Lowercase	0.80
MIN-K% PROB	**0.88**

Figure 3: AUC scores for detecting the validation set of copyrighted books on GPT-3.

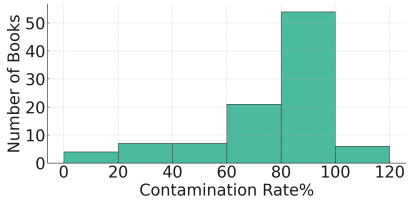


Figure 4: Distribution of detected contamination rate of 100 copyrighted books.

Table 2: Top 20 copyrighted books in GPT-3's pretraining data. The listed contamination rate represents the percentage of text excerpts from each book identified in the pretraining data.

Contamination %	Book Title	Author	Year
100	The Violin of Auschwitz	Maria Àngels Anglada	2010
100	North American Stadiums	Grady Chambers	2018
100	White Chappell Scarlet Tracings	Iain Sinclair	1987
100	Lost and Found	Alan Dean	2001
100	A Different City	Tanith Lee	2015
100	Our Lady of the Forest	David Guterson	2003
100	The Expelled	Mois Benarroch	2013
99	Blood Cursed	Archer Alex	2013
99	Genesis Code: A Thriller of the Near Future	Jamie Metzl	2014
99	The Sleepwalker's Guide to Dancing	Mira Jacob	2014
99	The Harlan Ellison Hornbook	Harlan Ellison	1990
99	The Book of Freedom	Paul Selig	2018
99	Three Strong Women	Marie NDiaye	2009
99	The Leadership Mind Switch: Rethinking How We Lead in the New World of Work	D. A. Benton, Kylie Wright-Ford	2017
99	Gold	Chris Cleave	2012
99	The Tower	Simon Clark	2005
98	Amazon	Bruce Parry	2009
98	Ain't It Time We Said Goodbye: The Rolling Stones on the Road to Exile	Robert Greenfield	2014
98	Page One	David Folkenflik	2011
98	Road of Bones: The Siege of Kohima 1944	Fergal Keane	2010

6 Case Study: Detecting Downstream Dataset Contamination

Assessing the leakage of downstream task data into pretraining corpora is an important issue, but it is challenging to address given the lack of access to pretraining datasets. In this section, we investigate the possibility of using MIN-K% PROB to detect information leakage and perform ablation studies to understand how various training factors impact detection difficulty. Specifically, we continually pretrain the 7B parameter LLaMA model (Touvron et al., 2023a) on pretraining data that have been purposefully contaminated with examples from the downstream task.

6.1 Experiments

Experimental setup. To simulate downstream task contamination that could occur in real-world settings, we create contaminated pretraining data by inserting examples from downstream tasks into a pretraining corpus. Specifically, we sample text from the RedPajama corpus (TogetherCompute, 2023) and insert formatted examples from the downstream datasets BoolQ (Clark et al., 2019), IMDB (Maas et al., 2011), Truthful QA (Lin et al., 2021), and Commonsense QA (Talmor et al., 2019) in contiguous segments at random positions in the uncontaminated text. We insert 200 (positive) examples from each of these datasets into the pretraining data while also isolating a set of 200 (negative) examples from each dataset that are known to be absent from the contaminated corpus. This creates a contaminated pretraining dataset containing 27 million tokens with 0.1% drawn from downstream datasets.

We evaluate the effectiveness of MIN-K% PROB at detecting leaked benchmark examples by computing AUC scores over these 400 examples on a LLaMA 7B model finetuned for one epoch on our contaminated pretraining data at a constant learning rate of 1e-4.

Main results. We present the main attack results in Table 3. We find that MIN-K% PROB outperforms all baselines. We report TPR@5%FPR in Table **??** in Appendix A, where MIN-K% PROB shows 12.2% improvement over the best baseline.

Table 3: AUC scores for detecting contaminant downstream examples. **Bold** shows the best AUC score within each column.

Method	BoolQ	Commonsense QA	IMDB	Truthful QA	Avg.
Neighbor	0.68	0.56	0.80	0.59	0.66
Zlib	0.76	0.63	0.71	0.63	0.68
Lowercase	0.74	0.61	0.79	0.56	0.68
PPL	0.89	0.78	0.97	0.71	0.84
MIN-K% PROB	**0.91**	**0.80**	**0.98**	**0.74**	**0.86**

6.2 Results and Analysis

The simulation with contaminated datasets allows us to perform ablation studies to empirically analyze the effects of *dataset size*, *frequency of data occurrence*, and *learning rate* on detection difficulty, as theorized in section 2.1. The empirical results largely align with and validate the theoretical framework proposed. In summary, we find that detection becomes more challenging as data occurrence and learning rate decreases, and the effect of dataset size on detection difficulty depends on whether the contaminants are outliers relative to the distribution of the pretraining data.

Pretraining dataset size. We construct contaminated datasets of 0.17M, 0.27M, 2.6M and 26M tokens by mixing fixed downstream examples (200 examples per downstream task) with varying amounts of RedPajama data, mimicking real-world pretraining. Despite the theory suggesting greater difficulty with more pretraining data, Figure 5a shows AUC scores counterintuitively increase with pre-training dataset size. This aligns with findings that LMs better memorize tail outliers (Feldman, 2020; Zhang et al., 2021). With more RedPajama tokens in the constructed dataset, downstream examples become more significant outliers. We hypothesize that their enhanced memorization likely enables easier detection with perplexity-based metrics.

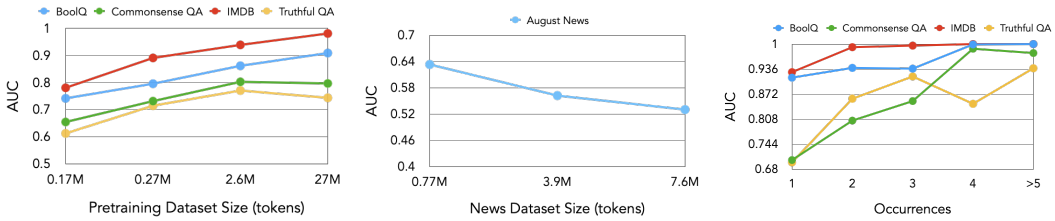
To verify the our hypothesis, we construct control data where contaminants are not outliers. We sample Real Time Data News August 2023[5], containing post-2023 news absent from LLaMA pretraining. We create three synthetic corpora by concatenating 1000, 5000 and 10000 examples from this corpus, hence creating corpora of sizes 0.77M, 3.9M and 7.6M tokens respecitvely. In each setting, we consider 100 of these examples to be contaminant (positive) examples and set aside another set of 100 examples from News August 2023 (negative). Figure 5b shows AUC scores decrease as the dataset size increases.

Detection of outlier contaminants like downstream examples gets easier as data size increases, since models effectively memorize long-tail samples. However, detecting general in-distribution samples from the pretraining data distribution gets harder with more data, following theoretical expectations.

Data occurrence. To study the relationship between detection difficulty and data occurrence, we construct a contaminated pretraining corpus by inserting multiple copies of each downstream data point into a pre-training corpus, where the occurrence of each example follows a Poisson distribution. We measure the relationship between the frequency of the example in the pretraining data and its AUC scores. Figure 5c shows that AUC scores positively correlates with the occurrence of examples.

Learning rate. We also study the effect of varying the learning rates used during pretraining on the detection statistics of the contaminant examples (see Table 4). We find that raising the learning rate from 10^{-5} to 10^{-4} increases AUC scores significantly in all the downstream tasks, implying

[5] https://huggingface.co/datasets/RealTimeData/News_August_2023



(a) Outlier contaminants, e.g., downstream examples, become easier to detect as dataset size increases.

(b) In-distribution contaminants, e.g., news articles, are harder to detect as dataset size increases.

(c) Contaminants that occur more frequently in the dataset are easier to detect.

Figure 5: We show the effect of contamination rate (expressed as a percentage of the total number of pretraining tokens) and occurrence frequency on the ease of detection of data contaminants using MIN-K% PROB.

that higher learning rates cause models to memorize their pretraining data more strongly. A more in-depth analysis in Table **??** in Appendix A demonstrates that a higher learning rate leads to more memorization rather than generalization for these downstream tasks.

Table 4: AUC scores for detecting contaminant downstream examples using two different learning rates. Detection becomes easier when higher learning rates are used during training. **Bold** shows the best AUC score within each column.

Learning rate	BoolQ	Commonsense QA	IMDB	LSAT QA	Truthful QA
1×10^{-5}	0.64	0.59	0.76	0.72	0.56
1×10^{-4}	**0.91**	**0.80**	**0.98**	**0.82**	**0.74**

7 Case Study: Privacy Auditing of Machine Unlearning

We also demonstrate that our proposed technique can effectively address the need for auditing machine unlearning, ensuring compliance with privacy regulations (Figure 6).

7.1 Backgrounding

The right to be forgotten and machine unlearning. In today's landscape of machine learning systems, it is imperative to uphold individuals' "right to be forgotten", a legal obligation outlined in regulations such as the General Data Protection Regulation (GDPR) (Voigt & Von dem Bussche, 2017) and the California Consumer Privacy Act (CCPA) (Legislature, 2018). This requirement allows users to request the removal of their data from trained models. To address this need, the concept of machine unlearning has emerged as a solution for purging data from machine learning models, and various machine unlearning methods have been introduced (Ginart et al., 2019; Liu et al., 2020; Wu et al., 2020; Bourtoule et al., 2021; Izzo et al., 2021; Sekhari et al., 2021; Gupta et al., 2021; Ye et al., 2022).

Recently, Eldan & Russinovich (2023) introduced a novel approach for performing machine unlearning on LLMs. This approach involves further fine-tuning the LLMs with alternative labels for specific tokens, effectively creating a modified version of the model that no longer contains the to-be-unlearned content. Specifically, the authors demonstrated the efficacy of this method using the `LLaMA2-7B-chat` model (Touvron et al., 2023b), showcasing its ability to "unlearn" information from the Harry Potter book series which results in the `LLaMA2-7B-WhoIsHarryPotter` model[6]. In this case study, we aim to assess whether this model successfully eliminates memorized content related to the Harry Potter series.

[6]Available at `https://huggingface.co/microsoft/Llama2-7b-WhoIsHarryPotter`.

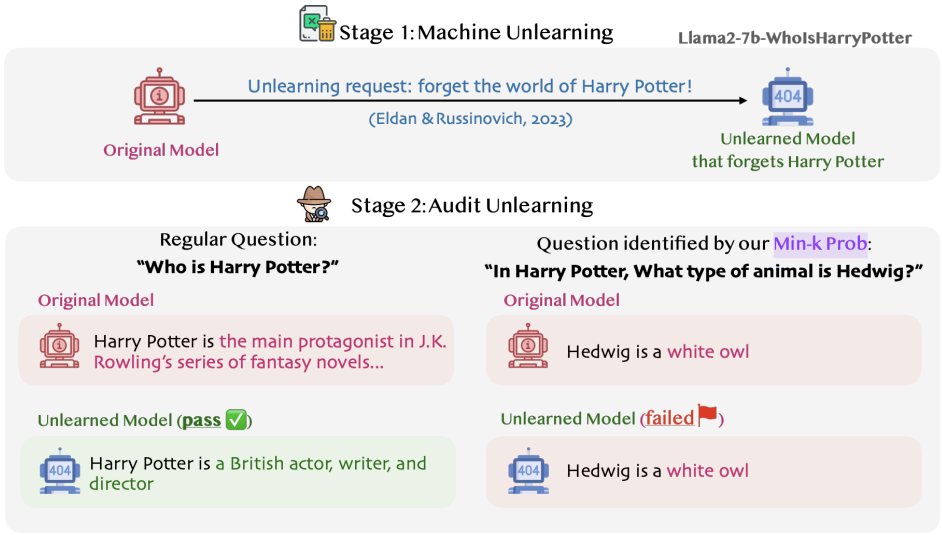


Figure 6: **Auditing machine unlearning with MIN-K% PROB**. Machine unlearning methods are designed to remove copyrighted and personal data from large language models. We use MIN-K% PROB to audit an unlearned LLM that has been trained to forget copyrighted books. However, we find that such a model can still output related copyrighted content.

7.2 Experiments

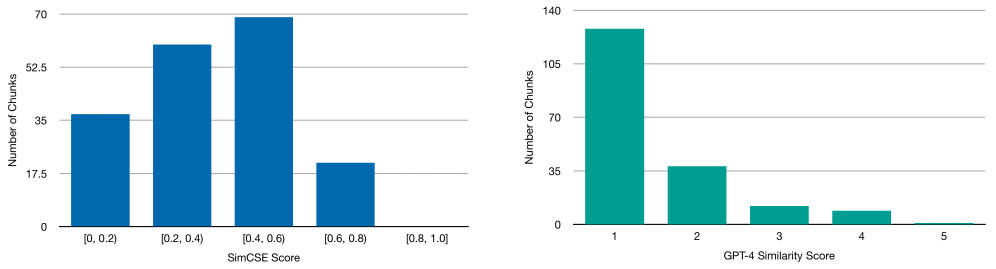
To extract the contents related to Harry Potter from the unlearned model, `LLaMA2-7B-WhoIsHarryPotter`, we consider two settings: *story completion* (§7.2.1) and *question answering* (§7.2.2). In *story completion*, we identify suspicious chunks from the original Harry Potter books using MIN-K% PROB. We then use the unlearned model to generate completions and compare them with the gold continuation. In *question answering*, we generate a series of questions related to Harry Potter using GPT-4 [7]. We filter these questions using MIN-K% PROB, and then use the unlearned model to produce answers. These answers are then compared with the gold answers generated by GPT-4 and subsequently verified by humans.

7.2.1 Story completion

Identifying suspicious texts using MIN-K% PROB. The process begins with the identification of suspicious chunks using our MIN-K% PROB metric. Firstly, we gather the plain text of Harry Potter Series 1 to 4 and segment these books into 512-word chunks, resulting in approximately 1000 chunks. We then compute the MIN-K% PROB scores for these chunks using both the `LLaMA2-7B-WhoIsHarryPotter` model and the original `LLaMA2-7B-chat` model. To identify chunks where the unlearning process may have failed at, we compare the MIN-K% PROB scores between the two models. If the ratio of the scores from the two models falls within the range of $(\frac{1}{1.15}, 1.15)$, we classify the chunk as a suspicious unlearn-failed chunk. This screening process identifies 188 such chunks. We also notice that using perplexity alone as the metric fails to identify any such chunk. We then test the `LLaMA2-7B-WhoIsHarryPotter` model with these suspicious chunks to assess its ability to complete the story. For each suspicious chunk, we prompt the model with its initial 200 words and use multinomial sampling to sample 20 model-generated continuations for each chunk.

Results We compare the completed stories with the ground truth storylines using both the SimCSE score (Gao et al., 2021) (which gives a similarity score from 0 to 1) and GPT-4 (where we prompt the model with the template in **??** to return a similarity score from 1 to 5, and a reason explaining the similarity). We can still find very similar completion with the original story. For example, 5.3% generated completions have greater and equal to 4 GPT score similarity to the gold completion. The distributions for these two scores of the suspicious chunks are shown in Section 7.2.1. Surprisingly, we find a considerable number of chunks whose auto-completions from the "unlearned" model closely resemble the original story: 10 chunks have a similarity score higher than or equal to 4 according to the GPT-4 evaluator. For instance, **??** showcases a few such examples, with all of them having

[7] OpenAI. `https://chat.openai.com/chat`



(a) SimCSE score

(b) GPT-scored similarity

Figure 7: Distribution of the SimCSE score (a) and GPT-scored similarity (b) between the original story and the completion by the `LLaMA2-7B-WhoIsHarryPotter` model.

Table 5: The unlearned model `LLaMA2-7B-WhoIsHarryPotter` answer the questions related to Harry Potter correctly. We manually cross-checked these responses against the Harry Potter book series for verification.

Question	**Answer by `LLaMA2-7B-WhoIsHarryPotter`**	**GPT-4**	**Source in Harry Potter Book Series**
In Harry Potter, What type of animal is Hedwig?	Hedwig is **a white owl**.	Hedwig is **a white owl**.	*"For Harry's birthday, Hagrid buys Harry a **snowy owl named Hedwig**." – Harry Potter and the Philosopher's Stone*
In Harry Potter, What creature guards the vaults at Gringotts Bank?	In the Harry Roper series by T.L. Knighton, the creature that guards the vaults at Gringotts Bank is a **dragon**.	The creature that guards the vaults at Gringotts Bank is a **dragon**.	*"With Harry and Griphook under the Invisibility Cloak, Hermione disguised as Bellatrix and Ron as a foreign wizard with bushy facial hair, the trio and Griphook Apparate to **Gringotts** ... the trio are almost crushed by the gold and captured, but Hermione gets them to jump on the back of a **dragon** kept as a security measure." – Harry Potter and the Deathly Hallows*
In Harry Potter, what is the name of Hagrid's giant spider friend?	Hagrid's giant spider friend is named **Aragog**.	**Aragog**	*"'**Aragog**!' it called. '**Aragog**!' And from the middle of the misty, domed web, **a spider the size of a small elephant** emerged, very slowly." – Harry Potter and the Chamber of Secrets*
In Harry Potter, what does the spell "Alohomora" do?	In the "Magic for Good" series by John G. Hartness, the spell "Alohomora" is a spell for **unlocking doors**.	**Unlocks doors**.	*"She grabbed Harry's wand, tapped the lock, and whispered, "Alohomora!" **The lock clicked and the door swung open** – they piled through it, shut it quickly..." – Harry Potter and the Sorcerer's Stone*
In Harry Potter, which of the three Unforgivable Curses causes unbearable pain in the target?	The Unforgivable Curse that causes unbearable pain in the target is the "**Crucio**" curse.	**Crucio**	*"'**Crucio**!' At once, the spider's legs bent in upon its body; it rolled over and began to twitch horribly, rocking from side to side. No sound came from it, but Harry was sure that if it could have given voice, it would have been screaming." – Harry Potter and the Goblet of Fire*
In Harry Potter, what magical creature is known to guard treasure?	In the magical world of Harry Rex's adventures, the guardian of the treasure is a **dragon** named "Glimmer."	**Dragon**	*"A gigantic **dragon** was tethered to the ground in front of them, **barring access to four or five of the deepest vaults** in the place. " – Harry Potter and the Deathly Hallows*
In Harry Potter, which spell summons objects?	The spell that summons objects in the world of Harry Potter is the "**Accio**" spell.	**Accio**	*"'**Accio! Accio! Accio!**' she shouted, and toffees **zoomed from all sorts of unlikely places**, including the lining of George's jacket..." – Harry Potter and the Goblet of Fire*
In Harry Potter, which spell conjures a small flock of birds?	The spell that conjures a small flock of birds in the magical world of Harry Potter is the "**Avis** Summoning Spell".	**Avis**	*'**Avis**!' The hornbeam wand let off a blast hike a gun, and **a number of small, twittering birds** flew out of the end and through the open window into the watery sunlight. – Harry Potter and the Goblet of Fire*

SimCSE scores exceeding 0.7. We further note that the study only uses Harry Potter books 1 to 4. Including the whole Harry Potter series (7 books) potentially will expose more unlearn-failed chunks.

7.2.2 Question answering

Selecting Harry Potter-related questions with MIN-K% PROB We generate 1000 questions related to Harry Potter by prompting GPT-4 with the query "Can you give me a list of questions and answers related to Harry Potter". Similar to identifying suspocious texts in story completion, we compare the MIN-K% PROB scores between the original and unlearned models and select questions

with the ratio falling within the range of $(\frac{1}{1.15}, 1.15)$, resulting in 103 questions. We use the unlearned model to generate answer given these questions, specifically employing multinomial sampling to sample 20 answers for each question.

Results We then compare the answers by the unlearned model (referred to as the "candidate") to those provided by GPT-4 (referred to as the "reference") using the ROUGE-L recall measure (Lin, 2004), which calculates the ratio: (# overlapping words between the candidate and reference) / (# words in the reference). A higher ROUGE-L recall value signifies a greater degree of overlap, which can indicate a higher likelihood of unlearning failure. Among the 103 selected questions, we observe an average ROUGE-L recall of 0.23. Conversely, for the unselected questions, the average ROUGE-L recall is 0.10. These findings underscore the capability of our MIN-K% PROB to identify potentially unsuccessful instances of unlearning.

Table 5 shows the selected questions related to Harry Potter that are answered correctly by the unlearned model `LLaMA2-7B-WhoIsHarryPotter` (with ROUGE-L recall being 1). We also verify the generated answers by cross-checking them against the Harry Potter series. These results suggest the knowledge about Harry Potter is not completely erased from the unlearned model.

8 Related Work

Membership inference attack in NLP. Membership Inference Attacks (MIAs) aim to determine whether an arbitrary sample is part of a given model's training data (Shokri et al., 2017; Yeom et al., 2018b). These attacks pose substantial privacy risks to individuals and often serve as a basis for more severe attacks, such as data reconstruction (Carlini et al., 2021; Gupta et al., 2022; Cummings et al., 2023). Due to its fundamental association with privacy risk, MIA has more recently found applications in quantifying privacy vulnerabilities within machine learning models and in verifying the accurate implementation of privacy-preserving mechanisms (Jayaraman & Evans, 2019; Jagielski et al., 2020; Zanella-Béguelin et al., 2020; Nasr et al., 2021; Huang et al., 2022; Nasr et al., 2023; Steinke et al., 2023). Initially applied to tabular and computer vision data, the concept of MIA has recently expanded into the realm of language-oriented tasks. However, this expansion has predominantly centered around finetuning data detection (Song & Shmatikov, 2019; Shejwalkar et al., 2021; Mahloujifar et al., 2021; Jagannatha et al., 2021; Mireshghallah et al., 2022b). Our work focuses on the application of MIA to pretraining data detection, an area that has received limited attention in previous research efforts.

Dataset contamination. The dataset contamination issue in LMs has gained attention recently since benchmark evaluation is undermined if evaluation examples are accidentally seen during pre-training. Brown et al. (2020b), Wei et al. (2022), and Du et al. (2022) consider an example contaminated if there is a 13-gram collision between the training data and evaluation example. Chowdhery et al. (2022) further improves this by deeming an example contaminated if 70% of its 8-grams appear in the training data. Touvron et al. (2023b) builds on these methods by extending the framework to tokenized inputs and judging a token to be contaminated if it appears in any token n-gram longer than 10 tokens. However, their methods require access to retraining corpora, which is largely unavailable for recent model releases. Other approaches try to detect contamination without access to pretraining corpora. Sainz et al. (2023) simply prompts ChatGPT to generate examples from a dataset by providing the dataset's name and split. They found that the models generate verbatim instances from NLP datasets. Golchin & Surdeanu (2023) extends this framework to extract more memorized instances by incorporating partial instance content into the prompt. Similarly, Weller et al. (2023) demonstrates the ability to extract memorized snippets from Wikipedia via prompting. While these methods study contamination in closed-sourced models, they cannot determine contamination on an instance level. Marone & Van Durme (2023) argues that model-developers should release training data membership testing tools accompanying their LLMs to remedy this. However, this is not yet widely practiced.

9 Conclusion

We present a pre-training data detection dataset WIKIMIA and a new approach MIN-K% PROB. Our approach uses the intuition that trained data tends to contain fewer outlier tokens with very low

probabilities compared to other baselines. Additionally, we verify the effectiveness of our approach in real-world setting, we perform two case studiies: detecting dataset contamination and published book detection. For dataset contamination, we observe empirical results aligning with theoretical predictions about how detection difficulty changes with dataset size, example frequency, and learning rate. Most strikingly, our book detection experiments provide strong evidence that GPT-3 models may have been trained on copyrighted books.

References

Raef Bassily, Vitaly Feldman, Cristóbal Guzmán, and Kunal Talwar. Stability of stochastic gradient descent on nonsmooth convex losses. *Advances in Neural Information Processing Systems*, 33: 4381–4391, 2020.

Stella Biderman, Hailey Schoelkopf, Quentin Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar van der Wal. Pythia: A suite for analyzing large language models across training and scaling, 2023.

Sid Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, Michael Pieler, USVSN Sai Prashanth, Shivanshu Purohit, Laria Reynolds, Jonathan Tow, Ben Wang, and Samuel Weinbach. GPT-NeoX-20B: An open-source autoregressive language model. In *Proceedings of the ACL Workshop on Challenges & Perspectives in Creating Large Language Models*, 2022. URL https://arxiv.org/abs/2204.06745.

Lucas Bourtoule, Varun Chandrasekaran, Christopher A Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. Machine unlearning. In *2021 IEEE Symposium on Security and Privacy (SP)*, pp. 141–159. IEEE, 2021.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901. Curran Associates, Inc., 2020a. URL https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020b.

Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, et al. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, pp. 2633–2650, 2021.

Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramer. Membership inference attacks from first principles. In *2022 IEEE Symposium on Security and Privacy (SP)*, pp. 1897–1914. IEEE, 2022.

Kent K Chang, Mackenzie Cramer, Sandeep Soni, and David Bamman. Speak, memory: An archaeology of books known to chatgpt/gpt-4. *arXiv preprint arXiv:2305.00118*, 2023.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. Boolq: Exploring the surprising difficulty of natural yes/no questions. In *NAACL*, 2019.

Rachel Cummings, Damien Desfontaines, David Evans, Roxana Geambasu, Matthew Jagielski, Yangsibo Huang, Peter Kairouz, Gautam Kamath, Sewoong Oh, Olga Ohrimenko, et al. Challenges towards the next frontier in privacy. *arXiv preprint arXiv:2304.06929*, 2023.

Nan Du, Yanping Huang, Andrew M Dai, Simon Tong, Dmitry Lepikhin, Yuanzhong Xu, Maxim Krikun, Yanqi Zhou, Adams Wei Yu, Orhan Firat, et al. Glam: Efficient scaling of language models with mixture-of-experts. In *International Conference on Machine Learning*, pp. 5547–5569. PMLR, 2022.

Ronen Eldan and Mark Russinovich. Who's Harry Potter? approximate unlearning in LLMs. *arXiv preprint arXiv:2310.02238*, 2023.

Vitaly Feldman. Does learning require memorization? a short tale about a long tail. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pp. 954–959, 2020.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. SimCSE: Simple contrastive learning of sentence embeddings. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2021.

Antonio Ginart, Melody Guan, Gregory Valiant, and James Y Zou. Making ai forget you: Data deletion in machine learning. *Advances in neural information processing systems*, 32, 2019.

Shahriar Golchin and Mihai Surdeanu. Time travel in llms: Tracing data contamination in large language models. *arXiv preprint arXiv:2308.08493*, 2023.

Samyak Gupta, Yangsibo Huang, Zexuan Zhong, Tianyu Gao, Kai Li, and Danqi Chen. Recovering private text in federated learning of language models. *Advances in Neural Information Processing Systems*, 35:8130–8143, 2022.

Varun Gupta, Christopher Jung, Seth Neel, Aaron Roth, Saeed Sharifi-Malvajerdi, and Chris Waites. Adaptive machine unlearning. *Advances in Neural Information Processing Systems*, 34:16319–16330, 2021.

Moritz Hardt, Ben Recht, and Yoram Singer. Train faster, generalize better: Stability of stochastic gradient descent. In *International conference on machine learning*, pp. 1225–1234. PMLR, 2016.

Yangsibo Huang, Chun-Yin Huang, Xiaoxiao Li, and Kai Li. A dataset auditing method for collaboratively trained machine learning models. *IEEE Transactions on Medical Imaging*, 2022.

Zachary Izzo, Mary Anne Smart, Kamalika Chaudhuri, and James Zou. Approximate data deletion from machine learning models. In *International Conference on Artificial Intelligence and Statistics*, pp. 2008–2016. PMLR, 2021.

Abhyuday Jagannatha, Bhanu Pratap Singh Rawat, and Hong Yu. Membership inference attack susceptibility of clinical language models. *arXiv preprint arXiv:2104.08305*, 2021.

Matthew Jagielski, Jonathan Ullman, and Alina Oprea. Auditing differentially private machine learning: How private is private sgd? *Advances in Neural Information Processing Systems*, 33: 22205–22216, 2020.

Bargav Jayaraman and David Evans. Evaluating differentially private machine learning in practice. In *28th USENIX Security Symposium (USENIX Security 19)*, pp. 1895–1912, 2019.

Nikhil Kandpal, Eric Wallace, and Colin Raffel. Deduplicating training data mitigates privacy risks in language models. In *International Conference on Machine Learning*, pp. 10697–10707. PMLR, 2022.

California State Legislature. California consumer privacy act, 2018. URL `https://oag.ca.gov/privacy/ccpa`.

Klas Leino and Matt Fredrikson. Stolen memories: Leveraging model memorization for calibrated {White-Box} membership inference. In *29th USENIX security symposium (USENIX Security 20)*, pp. 1605–1622, 2020.

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, 2021.

Gaoyang Liu, Xiaoqiang Ma, Yang Yang, Chen Wang, and Jiangchuan Liu. Federated unlearning. *arXiv preprint arXiv:2012.13891*, 2020.

Yunhui Long, Vincent Bindschaedler, Lei Wang, Diyue Bu, Xiaofeng Wang, Haixu Tang, Carl A Gunter, and Kai Chen. Understanding membership inferences on well-generalized learning models. *arXiv preprint arXiv:1802.04889*, 2018.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/P11-1015.

Inbal Magar and Roy Schwartz. Data contamination: From memorization to exploitation. *ArXiv*, abs/2203.08242, 2022. URL https://api.semanticscholar.org/CorpusID:247475929.

Saeed Mahloujifar, Huseyin A Inan, Melissa Chase, Esha Ghosh, and Marcello Hasegawa. Membership inference on word embedding and beyond. *arXiv preprint arXiv:2106.11384*, 2021.

Marc Marone and Benjamin Van Durme. Data portraits: Recording foundation model training data, 2023. URL https://arxiv.org/abs/2303.03919.

Justus Mattern, Fatemehsadat Mireshghallah, Zhijing Jin, Bernhard Schoelkopf, Mrinmaya Sachan, and Taylor Berg-Kirkpatrick. Membership inference attacks against language models via neighbourhood comparison. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 11330–11343, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.719. URL https://aclanthology.org/2023.findings-acl.719.

Sewon Min, Suchin Gururangan, Eric Wallace, Hannaneh Hajishirzi, Noah A Smith, and Luke Zettlemoyer. Silo language models: Isolating legal risk in a nonparametric datastore. *arXiv preprint arXiv:2308.04430*, 2023.

Fatemehsadat Mireshghallah, Kartik Goyal, Archit Uniyal, Taylor Berg-Kirkpatrick, and Reza Shokri. Quantifying privacy risks of masked language models using membership inference attacks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 8332–8347, Abu Dhabi, United Arab Emirates, December 2022a. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.570. URL https://aclanthology.org/2022.emnlp-main.570.

Fatemehsadat Mireshghallah, Kartik Goyal, Archit Uniyal, Taylor Berg-Kirkpatrick, and Reza Shokri. Quantifying privacy risks of masked language models using membership inference attacks. *arXiv preprint arXiv:2203.03929*, 2022b.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D. Manning, and Chelsea Finn. Detectgpt: Zero-shot machine-generated text detection using probability curvature, 2023. URL https://arxiv.org/abs/2301.11305.

Maximilian Mozes, Xuanli He, Bennett Kleinberg, and Lewis D. Griffin. Use of llms for illicit purposes: Threats, prevention measures, and vulnerabilities, 2023.

Arvind Narayanan. Gpt-4 and professional benchmarks: the wrong answer to the wrong question, 2023. URL https://www.aisnakeoil.com/p/gpt-4-and-professional-benchmarks.

Milad Nasr, Shuang Songi, Abhradeep Thakurta, Nicolas Papernot, and Nicholas Carlin. Adversary instantiation: Lower bounds for differentially private machine learning. In *2021 IEEE Symposium on security and privacy (SP)*, pp. 866–882. IEEE, 2021.

Milad Nasr, Jamie Hayes, Thomas Steinke, Borja Balle, Florian Tramèr, Matthew Jagielski, Nicholas Carlini, and Andreas Terzis. Tight auditing of differentially private machine learning. *arXiv preprint arXiv:2302.07956*, 2023.

OpenAI. Gpt-4 technical report, 2023.

Oscar Sainz, Jon Ander Campos, Iker García-Ferrero, Julen Etxaniz, and Eneko Agirre. Did chatgpt cheat on your test?, 2023. URL https://hitz-zentroa.github.io/lm-contamination/blog/.

Ayush Sekhari, Jayadev Acharya, Gautam Kamath, and Ananda Theertha Suresh. Remember what you want to forget: Algorithms for machine unlearning. *Advances in Neural Information Processing Systems*, 34:18075–18086, 2021.

Virat Shejwalkar, Huseyin A Inan, Amir Houmansadr, and Robert Sim. Membership inference attacks against NLP classification models. In *NeurIPS 2021 Workshop Privacy in Machine Learning*, 2021. URL https://openreview.net/forum?id=74lwg5oxheC.

R. Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 3–18, 2016.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*, pp. 3–18. IEEE, 2017.

Congzheng Song and Vitaly Shmatikov. Auditing data provenance in text-generation models. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 196–206, 2019.

Thomas Steinke, Milad Nasr, and Matthew Jagielski. Privacy auditing with one (1) training run. *arXiv preprint arXiv:2305.08846*, 2023.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. CommonsenseQA: A question answering challenge targeting commonsense knowledge. 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)*, pp. 4149–4158, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1421. URL https://aclanthology.org/N19-1421.

TogetherCompute. Redpajama: An open source recipe to reproduce llama training dataset, 2023. URL https://github.com/togethercomputer/RedPajama-Data.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023a.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023b.

Paul Voigt and Axel Von dem Bussche. The eu general data protection regulation (gdpr). *A Practical Guide, 1st Ed., Cham: Springer International Publishing*, 10(3152676):10–5555, 2017.

Lauren Watson, Chuan Guo, Graham Cormode, and Alexandre Sablayrolles. On the importance of difficulty calibration in membership inference attacks. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=3eIrli0TwQ.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=gEZrGCozdqR.

Orion Weller, Marc Marone, Nathaniel Weir, Dawn Lawrie, Daniel Khashabi, and Benjamin Van Durme. "according to ..." prompting language models improves quoting from pre-training data, 2023.

Yinjun Wu, Edgar Dobriban, and Susan Davidson. Deltagrad: Rapid retraining of machine learning models. In *International Conference on Machine Learning*, pp. 10355–10366. PMLR, 2020.

Jingwen Ye, Yifang Fu, Jie Song, Xingyi Yang, Songhua Liu, Xin Jin, Mingli Song, and Xinchao Wang. Learning with recoverable forgetting. In *European Conference on Computer Vision*, pp. 87–103. Springer, 2022.

Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st Computer Security Foundations Symposium (CSF)*, pp. 268–282, 2018a. doi: 10.1109/CSF.2018.00027.

Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st computer security foundations symposium (CSF)*, pp. 268–282. IEEE, 2018b.

Santiago Zanella-Béguelin, Lukas Wutschitz, Shruti Tople, Victor Rühle, Andrew Paverd, Olga Ohrimenko, Boris Köpf, and Marc Brockschmidt. Analyzing information leakage of updates to natural language models. In *Proceedings of the 2020 ACM SIGSAC conference on computer and communications security*, pp. 363–375, 2020.

Chiyuan Zhang, Daphne Ippolito, Katherine Lee, Matthew Jagielski, Florian Tramèr, and Nicholas Carlini. Counterfactual memorization in neural language models. *arXiv preprint arXiv:2112.12938*, 2021.