Perturb Your Data: Paraphrase-Guided Training Data Watermarking

Pranav Shetty, Mirazul Haque,
Petr Babkin, Zhiqiang Ma, Xiaomo Liu, Manuela Veloso
JPMorgan AI Research
{first.last}@jpmchase.com

Abstract

Training data detection is critical for enforcing copyright and data licensing, as Large Language Models (LLM) are trained on massive text corpora scraped from the internet. We present SPECTRA, a watermarking approach that makes training data reliably detectable even when it comprises less than 0.001% of the training corpus. SPECTRA works by paraphrasing text using an LLM and assigning a score based on how likely each paraphrase is, according to a separate scoring model. A paraphrase is chosen so that its score closely matches that of the original text, to avoid introducing any distribution shifts. To test whether a suspect model has been trained on the watermarked data, we compare its token probabilities against those of the scoring model. We demonstrate that SPECTRA achieves a consistent p-value gap of over nine orders of magnitude when detecting data used for training versus data not used for training, which is greater than all baselines tested. SPECTRA equips data owners with a scalable, deploy-before-release watermark that survives even large-scale LLM training.

Extended version — https://arxiv.org/abs/2512.17075

Introduction

Contemporary large language models (LLMs) utilize extensive datasets sourced from the internet for pretraining, which lead to their general-purpose abilities, but may include content scraped without permission. Although these large corpora are necessary for the emergent abilities of LLMs, this practice raises several ethical and legal concerns. Firstly, the utilization of copyrighted material in the training process may violate its licensing terms. Secondly, the incorporation of widely recognized benchmarking datasets during pretraining could compromise the integrity of model evaluation. Many open-weight models are released without disclosing their training data, which leaves model end-users vulnerable to liability for damages and can hinder the adoption of open-source models.

Several recent lawsuits have focused on the unauthorized use of pay-walled data that was used for training models (Stempel 2023; Brittain 2024). As more and more content is consumed online through various intermediate LLMs like ChatGPT, it becomes vital to ensure that content creators are appropriately incentivized to produce new content. Otherwise, the data collection practices of most LLM providers may lead to an "extractive dead end" (Rosenblat, O'Reilly, and Strauss 2025). Consequently, there is an important need to detect unauthorized use of data for training.

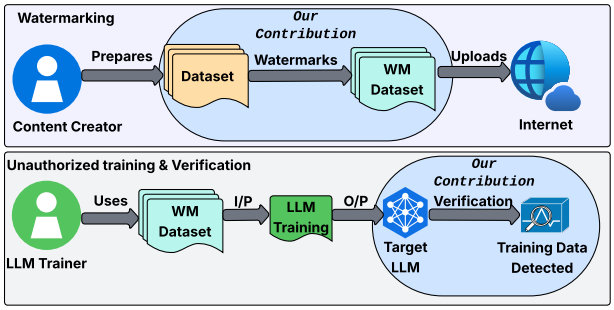

Figure 1: Overview of our problem setting. I/P and O/P refer to input and output, respectively.

Many recent techniques have been developed to detect training data (Yeom et al. 2018; Zhang et al. 2025, 2024). Membership Inference Attack (MIA) techniques build on the idea that training leads to changes in the probabilities output by the model, which is measured by comparing these probabilities (MIA scores) for training data (also called member data) against a held-out or non-member dataset from the same domain that was not used for training. A content creator must provide the suspect data and held-out data, which can be used to test a target model. However, MIA techniques are sensitive to small distribution shifts between the suspect and held-out data that can lead to spurious performance (Duan et al. 2024; Maini et al. 2024; Zhao et al. 2025).

Given the limitations of standard MIA techniques, it is necessary to find alternative ways through which content creators can protect their content. STAMP (Rastogi, Maini, and Pruthi 2025) is a recent effort in this direction that watermarks text using the KGW scheme (Kirchenbauer et al. 2023) multiple times with different keys, with one version made public and the rest kept private by the content cre-

ator. If their data is used for training, content creators can compare the perplexity of the public rephrases against the private rephrases to establish membership (included in the training data). In contrast to MIA, which detects the membership of each document, STAMP performs inference over the entire dataset, enabling it to be robust to noise in individual examples. STAMP requires storing a large number of private rephrases for each document. This method also requires access to the decoding layers of an LLM to generate watermarked text, which may be inaccessible due to the large amount of GPU resources required. The p-values between member and non-member datasets are at most three orders of magnitude apart in STAMP, which we argue is not sufficient for such a task with significant legal implications.

To address these limitations, we introduce **S**core sampled re**P**hrasing to det**EC**t **TRA**ining data (SPECTRA) (Figure 1). We assume that a content creator creates textual content and, before making it publicly available, watermarks a portion of it using SPECTRA. Such content might be openly accessible, placed behind a paywall, or protected by licenses explicitly prohibiting web scraping for model training. If the content creator suspects a particular model of unauthorized use of their data, they or an authorized intermediary can conduct a statistical test to confirm whether the watermarked data was indeed part of that model's training data.

The SPECTRA approach can be divided into two phases: watermarking and verification. During *watermarking*, SPECTRA generates several paraphrases using an LLM without needing to modify its decoding process. A score that provides a membership signal is computed over the paraphrases using a scoring model that has not been trained on this data. Specifically, we compute the Min-K%++ (Zhang et al. 2025) score, which is the normalized log probability averaged over the lowest $K\%$ tokens (highest surprisal tokens) in a document. Our key insight is that existing MIA scores are designed to measure changes in the loss surface between a model before and after it is trained on specific data, but in practice, we seldom have access to the model before training. We employ the scoring model as a proxy for this initial (pre-trained) state, allowing us to detect changes attributable to training reliably. To avoid introducing systematic bias, paraphrases are carefully selected using a sampling strategy that we designed, such that their Min-K%++ scores remain close to the original document's score. This constraint ensures that the watermarking procedure itself does not introduce any false-positive membership signals. In the *verification* phase, we construct a statistical test in which the scores from the Min-K%++ scores from the scoring model are compared against the scores from the suspected model to obtain a p-value. Unlike other MIA methods, SPECTRA does not require a non-member dataset. We find that SPECTRA yields a statistically significant result for identifying membership in all datasets used for training, without yielding false positives. SPECTRA will equip content creators with the tools to enforce their intended data-use policies.

Our contributions are as follows:

1. We show that SPECTRA can be used to watermark pre-training data and that the watermark can be measured after continued pre-training with 5 billion tokens. Each of our datasets constitute less than 0.001% of the training corpus.
2. We design a strategy to sample paraphrases that outperforms other sampling approaches, such as random sampling or selecting paraphrases with the maximum Min-K%++ score.
3. We benchmark against existing methods for detecting training data and find that SPECTRA is the only one that yields a statistically significant result to identify membership for all datasets tested. Moreover, SPECTRA gave the largest change in p-values between non-members and members, yielding a consistent difference of at least nine orders of magnitude across datasets using 500 samples each.

Related Work

Membership Inference Attack (MIA). In the context of LLMs, recent studies have proposed several Membership Inference Attack methods that use scores derived from the log probabilities output by the model to differentiate member data (used during training) from non-member data (not used during training). The datasets employed to evaluate these methods, such as WikiMIA (Shi et al. 2023) and PatentMIA (Zhang et al. 2024), were constructed by collecting data published before (member) and after (non-member) the LLM's knowledge cutoff date. It was later observed that these MIA techniques primarily detected temporal artifacts. Subsequent work showed that when member and non-member datasets were sampled homogeneously—thereby removing temporal signals—all tested MIA methods performed no better than random classifiers (Duan et al. 2024; Maini et al. 2024; Das, Zhang, and Trantèr 2025).

Data Watermarks. Data watermarks are modifications made to a text to make it more detectable when used to train a model. Wei, Wang, and Jia (2024) insert hash strings into a model and show that models trained on such hashes occurring multiple times in the dataset memorize the hash. Other works have proposed backdoor attacks that insert carefully picked tokens into the text, which can be detected post-training by using a secret prompt (Bouaziz et al. 2025; Yan, Gupta, and Ren 2022). The assumption underlying these works is that LLM trainers collect large quantities of data from the internet and are likely to collect data containing such watermarks. The challenge is that such watermarks affect the meaning and readability of the text and are thus not suitable when the text is meant for human readers. More recently, STAMP addresses this issue by rephrasing text from instruction tuning benchmarks using the Llama3-70b model and watermarking the text using the KGW scheme.

Dataset Inference. In contrast to MIA, Maini et al. (2024) propose Dataset Inference (DI) where the goal is not to obtain accurate labels over every document in the dataset but to obtain a measure of confidence over the entire dataset being tested. As LLM trainers tend to scrape each source comprehensively, it is likely that related documents from a single source will all be used for training. Using multiple docu-

ments also enhances the signal available for detecting membership and makes the inference less susceptible to noise due to outliers. Our work takes inspiration from this.

Problem Setup

Consider a content creator who possesses a dataset D, which they wish to make available online. D could consist of articles from sources such as news providers or academic publishers. The creator, however, wishes to prevent unauthorized use of this dataset for training LLMs.

To enable detection of unauthorized use, the creator applies a watermarking procedure W to transform the original dataset D into a modified dataset $D' = W(D)$. The creator retains D and only publishes D'.

After publication, the content creator may seek to test whether a particular target model M has been trained on the watermarked dataset D'. We assume a grey-box setting, where the model M is queried and provides log probabilities over tokens, but the model weights and architecture are not necessarily known. This scenario is typical of open-weight models, some of which are used commercially. For closed-source models, testing may be facilitated through a neutral third-party arbiter with grey-box access (say, a court-appointed arbiter). Given D, D' and grey-box access to M, the content creator or arbiter applies a statistical test T to determine if the model M was trained on D'. Note that this procedure detects membership of the entire dataset and not each document, which can be noisier.

The research question we address is: How can we optimally design the watermarking procedure W and the statistical test T such that:

1. If the model M is indeed trained on the watermarked dataset D', then T reliably identifies this fact with high confidence.
2. Conversely, if M is *not* trained on D', then T does not yield false-positive outcomes.

In the sections that follow, we propose a method to achieve these objectives and empirically validate its efficacy.

Training Data Detection Signals

Some common scores in the literature that are computed to determine membership in the training data of a model are described below. In our scenario, M is an autoregressive language model that generates a probability distribution for each subsequent token, denoted as $P(x_t \mid x_{<t}; M)$.

1. **Loss** (Yeom et al. 2018): This method relies on measuring the loss of a given target sequence x under the model M. The membership inference score is directly defined as: $f(x; M) = L(x; M)$ where $L(x; M)$ denotes the negative log-likelihood (loss) of the target sequence according to the model M.
2. **Min-K%** (Shi et al. 2023): This method focuses specifically on the $K\%$ of tokens that have the lowest likelihood under the model M. The membership inference score is computed as the average log probability over these tokens.
3. **Min-K%++** (Zhang et al. 2025) Min-K%++ enhances the original Min-K% score by incorporating normalization relative to the mean and variance of token log probabilities.

We focus on Min-K%++ in this work, which we describe in greater detail next.

Min-K%++ score

Formally, given an autoregressive model M and a token sequence $x = (x_1, x_2, \ldots, x_n)$, define the token-level normalized log probability as:

$$z(x_t; M) = \frac{\log P(x_t \mid x_{<t}; M) - \mu_{x_{<t}}}{\sigma_{x_{<t}}}$$

where

$$\mu_{x_{<t}} = \mathbb{E}_{z \sim P(\cdot \mid x_{<t}; M)}[\log P(z \mid x_{<t}; M)],$$

$$\sigma_{x_{<t}} = \sqrt{\mathbb{E}_{z \sim P(\cdot \mid x_{<t}; M)}\left[(\log P(z \mid x_{<t}; M) - \mu_{x_{<t}})^2\right]}.$$

Here, $\mu_{x_{<t}}$ represents the expectation of the log probability distribution for the next token given the prefix $x_{<t}$, and $\sigma_{x_{<t}}$ denotes the corresponding standard deviation.

The Min-K%++ score for a sequence x is defined as the average of the normalized log probabilities $z(x_t; M)$ over the $K\%$ of tokens in the sequence with the lowest values (indicating highest surprisal):

$$f_{\text{Min-K\%++}}(x; M) = \frac{1}{|\text{min-K}(x)|} \sum_{x_t \in \text{min-K}(x)} z(x_t; M).$$

This normalization allows Min-K%++ to better distinguish sequences that are part of the training data from those that are not, by highlighting the relative surprisal of the most unlikely tokens while making it robust to absolute probability shifts across tokens. Critically, prior work (Zhang et al. 2025) has shown that the Min-K%++ score theoretically corresponds to measuring the negative trace of the Hessian of the log-likelihood $log\ P(x_t \mid x_{<t})$. Intuitively, training via maximum-likelihood directly reduces the curvature (Hessian trace) of the loss landscape at training examples, thereby causing their corresponding Min-K%++ scores to increase. We evaluate the performance of various MIA scores on three datasets (Table 1) that were not used during the pretraining of Pythia models. When the datasets, each containing 500 samples (at most 512 tokens each), are used for training with an additional 500 million text tokens, Min-K%++ has the best performance among all methods. However, as reported by prior work (Duan et al. 2024), the effectiveness of Min-K%++ diminishes significantly at larger scales of training (5 billion tokens), dropping to performance indistinguishable from random. Based on these observations, we hypothesize that while Min-K%++ inherently captures strong training signals, its effectiveness at large scales of training may be hindered by distributional homogeneity when measured against non-member data, and that changing the reference point for measurement would enable us to capture a stronger training signal.

[1] See appendix for more details

	Metrics	Wiki	HN	PubMed
Datasets +500 million tokens	Loss	0.71	0.73	0.63
	DC-PDD[1]	0.77	0.79	0.64
	Min-k %	0.76	0.79	0.65
	Min-K%++	**0.85**	**0.84**	**0.72**
Datasets +5 billion tokens	Loss	0.55	0.54	0.52
	DC-PDD	0.55	0.52	0.50
	Min-k %	0.56	0.55	0.52
	Min-K%++	0.55	0.55	0.51

Table 1: ROC-AUC of classifying training data used for continued pretraining of a Pythia 410m model against a held-out dataset from the same domain. The ROC-AUC is computed using the MIA scores below. 500 million or 5 billion tokens are used during continued pretraining.

Watermarking with SPECTRA

This phase takes place after a content creator writes their content and before they publish it. To watermark a dataset, we generate multiple paraphrases of each document using a large language model and compute their Min-K%++ scores using a separate scoring model that has not been trained on the dataset being watermarked. This scoring model approximates the pre-training state of the target model. In practice, such a model is easy to find as D and D' are unpublished at the time of scoring. One can pick an open-source model with a knowledge cut-off prior to the release date of the dataset. We sample one paraphrase as the watermarked sample according to Algorithm 1.

The sampling favors paraphrases with scores close to the original score. For each side (above or below the original), we define a categorical distribution over candidate indices using weights proportional to $exp(-\alpha|r_{ij}-1|)$ where $\alpha > 0$ controls the sharpness of the distribution. A higher α causes the algorithm to favor paraphrases with scores close to the original more strongly. In practice, we pick the largest α that does not lead to numerical underflow issues ($\alpha = 100$).

If both above- and below-original paraphrases are available, the side is chosen probabilistically in inverse proportion to how often each type appears globally across the dataset, helping to avoid systematic score shifts. This sampling strategy ensures that the distribution of scores for the watermarked dataset remains similar to the original, reducing the likelihood of false positives when testing against models not trained on D'. Importantly, because the Min-K%++ scores of watermarked text tend to increase after training, the score distribution after training becomes distinguishable from the pre-training state. Notably, if paraphrases were consistently selected only from the high-score side, it would create a detectable signature even without training—leading to false positives. SPECTRA avoids this by balancing selection, ensuring a reliable signal only when training has occurred.

Algorithm 1: Sampling paraphrases

Input: Original scores $\{s_i^{(0)}\}_{i=1}^N$, paraphrased scores $\{S_i = \{s_{i1}, \ldots, s_{im}\}\}_{i=1}^N$, paraphrases $\{T_i = \{t_{i1}, \ldots, t_{im}\}\}_{i=1}^N$, parameter $\alpha = 100$
Output: Sampled paraphrases $\{t_{ij_i}\}_{i=1}^N$

1. Pre-computation: define $r_{ij} = s_{ij}/s_i^{(0)}$, then $\mathcal{A} = \{i : r_{ij} < 1, \forall j\}$, $\mathcal{B} = \{i : r_{ij} > 1, \forall j\}$.
2. Global side-balance:
   $\pi_+ = \begin{cases} 0.5 & \text{if } |\mathcal{A}| + |\mathcal{B}| = 0 \\ \frac{|\mathcal{A}|}{|\mathcal{A}|+|\mathcal{B}|} & \text{otherwise} \end{cases}$,
   $\pi_- = 1 - \pi_+$.
3. For each datapoint $i = 1, \ldots, N$:
   (a) Partition: $\mathcal{R}_i^{(-)} = \{j : r_{ij} < 1\}$, $\mathcal{R}_i^{(+)} = \{j : r_{ij} > 1\}$.
   (b) Side s_i: if one set empty, choose the other; else sample from $\{+, -\}$ w.p. π_+, π_-.
   (c) Let $\mathcal{R} = \mathcal{R}_i^{(s_i)}$.
   (d) Weights: $w_{ij} = \exp(-\alpha|r_{ij} - 1|)$ for $j \in \mathcal{R}$.
   (e) Normalize: $w_{ij} \leftarrow w_{ij} / \sum_{k \in \mathcal{R}} w_{ik}$.
   (f) Sample j_i from categorical$\{w_{ij}\}_{j \in \mathcal{R}}$.
4. Return $\{t_{ij_i}\}_{i=1}^N$.

Verification with SPECTRA

During this phase, the content is released to the public and it is suspected that the data may have been used in an unauthorized manner for training. Given an original document $x \in D$ and its watermarked counterpart $x' \in D'$, we define the score ratio under a model M as

$$r(x, x'; M) = \frac{f_{\text{Min-K\%++}}(x'; M)}{f_{\text{Min-K\%++}}(x; M)}.$$

Let M_S denote the scoring model (not trained on D'), M_T denote the target model (potentially trained on D'), and M_U denote the checkpoint of M_T before it was trained on D'. Given that Min-K%++ scores rise after training, we can write $\mathbb{E}_{x' \in D'}[f_{\text{Min-K\%++}}(x'; M_T)] > \mathbb{E}_{x' \in D'}[f_{\text{Min-K\%++}}(x'; M_U)]$. However, in practice, we do not have access to M_U and so we approximate it using M_S. However, because M_S and M_T may differ in architecture or baseline predictions, we normalize each watermarked score by the corresponding original document score to allow meaningful comparisons between the two models. Thus, under the null hypothesis H_0, the ratio of scores under M_T is equal to that under M_S when D' is not used for training M_T, i.e.,

$$H_0 : \mathbb{E}_{x,x'}[r(x, x'; M_T)] = \mathbb{E}_{x,x'}[r(x, x'; M_S)].$$

where the expectation is taken over $x \in D$ and $x' \in D'$. The alternate hypothesis H_1 states that the ratio of scores under the target model is lower relative to the scoring model when D' was used for training, i.e.,

$$H_1 : \mathbb{E}_{x,x'}[r(x, x'; M_T)] < \mathbb{E}_{x,x'}[r(x, x'; M_S)].$$

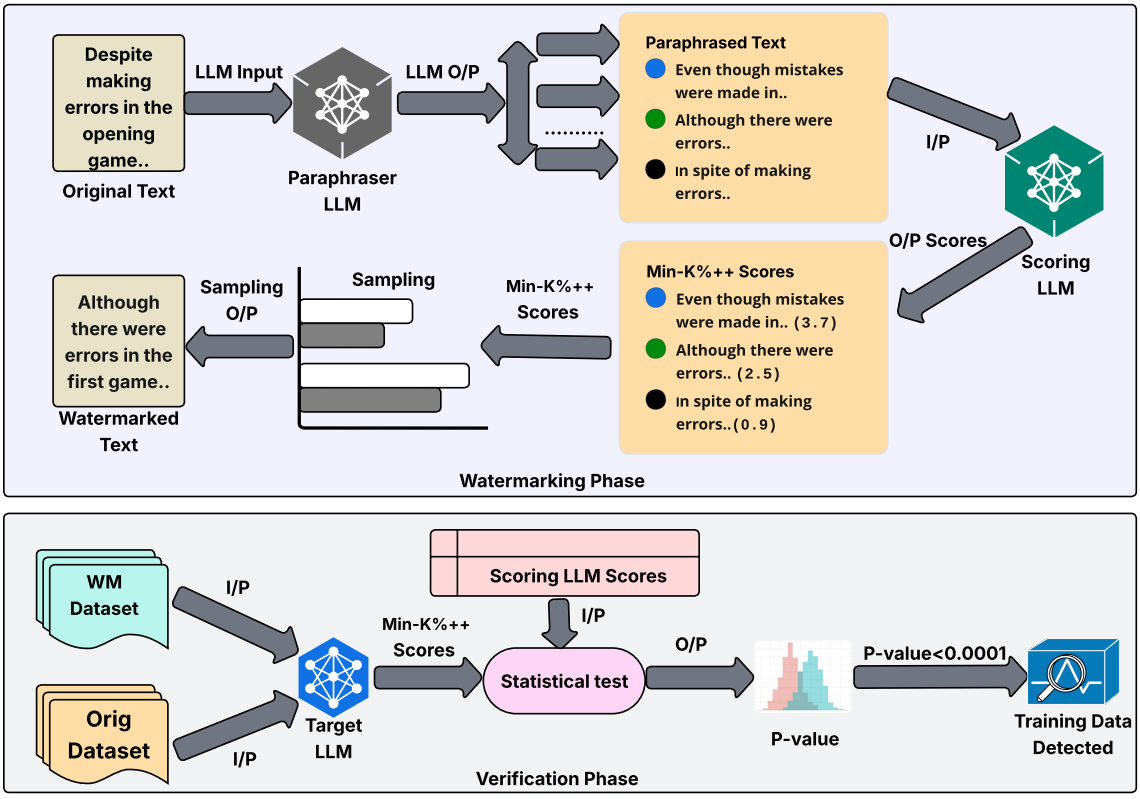

Figure 2: Overview of SPECTRA. **Watermarking Phase**: We use an LLM to generate multiple paraphrases of the original text. We sample one paraphrase that has a Min-K%++ score close to the original text. **Verification Phase**: Given a target LLM suspected of being trained on the watermarked data, we compute the Min-K%++ scores of the watermarked and original data and compare against the scores previously generated by the scoring model. Membership is detected through a paired t-test.

where the expectation is taken over $x \in D$ and $x' \in D'$. Note that the inequality flips sign here as Min-K%++ values in the denominator are always negative. We test these hypotheses by computing these ratios for all pairs (x, x') in each dataset and performing a 1-sided paired t-test. A low p-value would indicate rejection of the null hypothesis H_0, providing statistically significant evidence that the target model M_T has indeed been trained on the watermarked dataset D'.

Results

Datasets

We ensure that the datasets we use for training have not previously been used to train our target model of interest. Thus we are limited to training models where the training data is known transparently. Two prominent model families that meet this criterion are the Pythia models and the OLMo (Groeneveld et al. 2024) models along with their corresponding training datasets, The Pile and Dolma.

1. **The Pile**: We use the deduplicated subsets of the Pile (Gao et al. 2020; Duan et al. 2024) from the domains of Wikipedia (Wiki), Hackernews (HN), and Pubmed Central abstracts (PubMed) that were held out from training.
2. **Dolma**: We use the PeS2o held-out subset of Dolma obtained from Paloma (Soldaini et al. 2024; Magnusson et al. 2024). All text in this subset was released after the release date of the Pile, making it non-member for Pythia models.

Models

The watermarking pipeline and evaluation consists of 3 different types of models:

1. **Paraphraser model**: We use the Llama 3.1-405b model and generate 10 paraphrases per document (Grattafiori et al. 2024).
2. **Scoring model**: We use the Pythia 2.8b-deduped model for the Pile datasets. We use a model that is known not to have been trained on our datasets, as otherwise, the distribution of the Min-K%++ scores would shift higher. Pythia 2.8b-deduped has different weights but is from the same model family as our target model, i.e., Pythia 410m. Consequently, for PeS2o we use the OLMo-1b model to

investigate the effect of using a different model architecture between the scoring and target model.

3. **Target model**: This is the model that we suspect has been trained using D'. We use the vanilla Pythia 410m model and Pythia 410m on which we do continued pretraining using watermarked datasets, as target models. During continued pretraining, in addition to the watermarked text, we sample 5 billion tokens of text from the Common Pile dataset (Kandpal et al. 2025).

Baselines

We adopt the following baselines.

1. **Maximum**: pick the paraphrase with the highest Min-K%++ score.
2. **Random**: pick one of the paraphrases randomly.

Additionally, **STAMP** and **LLM-DI** are described in the appendix.

Evaluation of Watermarking

We measure statistical significance (p-values) for detecting the watermark in each dataset. Specifically, we compute member p-values from the Pythia-410m model trained on the watermarked datasets and non-member p-values from the original Pythia 410m model that has not encountered the watermarked data during training.

We see from the results (Table 2) that SPECTRA is the *only one* that correctly detects membership for each dataset in the study under a threshold of $p < 10^{-4}$. Under a naïve approach of selecting paraphrases that maximize the Min-K%++ shift, the resulting pre-training shift is so large that subsequent training does not further amplify it, making pre- and post-training distributions indistinguishable and thus undetectable in practice. The random baseline also fails as non-member Wiki results in false positives, while with PeS2o, it fails to detect membership. This indicates that the sampling strategy employed for paraphrases in SPECTRA is crucial to ensuring its performance. SPECTRA consistently has a high ratio of p-value between member data and non-member data ($> 10^9$) which is higher than the baselines. Notably, SPECTRA correctly detects membership for PeS2o, demonstrating that SPECTRA remains effective even as the scoring model and target model architecture differ. As suggested in Huh et al. (2024), different LLMs follow similar training objectives, and as the amount of training data and tasks gets scaled up, the space of acceptable representations narrows dramatically, leading to similar learned representations.

In contrast, STAMP and LLM-DI fail to reliably detect members under our strict threshold ($p < 10^{-4}$). Although these methods achieve significance for certain datasets when adopting a more permissive threshold $p < 0.05$, we argue that due to the significant implications of falsely identifying datasets as training data in LLMs, a more stringent threshold is necessary and justified.

Validating Quality of Paraphrases

We validate the quality of paraphrasing by using the P-SP metric (Wieting et al. 2022). The P-SP metric is widely used to measure paraphrasing quality (Rastogi, Maini, and Pruthi 2025; Krishna et al. 2023). For a human-generated paraphrase, the average P-SP is 0.78 (Krishna et al. 2023). Except for Hackernews, all watermarked datasets had a P-SP score above 0.88 (Table 3). The lower scores for Hackernews are explored next.

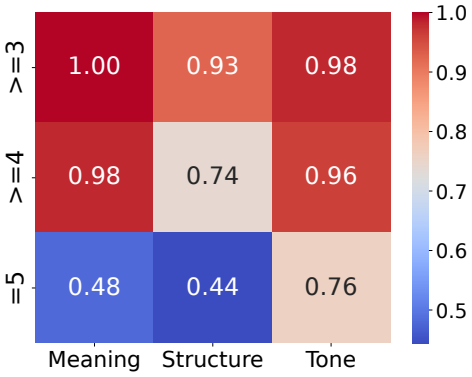

Figure 3: Heatmap showing fraction of evaluator scores for which the paraphrases received a rating ≥ 3.

Human evaluation. We randomly select 54 watermarked documents from our datasets and distribute them among four evaluators. This distribution ensures that each evaluator reviews 27 documents, with each document being assessed by two different evaluators. The evaluation focuses on whether the paraphrasing preserves the (i) meaning, (ii) structure, and (iii) author tone of the original text. Evaluators evaluate on a Likert scale (Likert 1932) of 1-5, with 5 being the best and 1 the worst.

Figure 3 shows the fraction of points that achieved a mean evaluator score of ≥ 3. While the mean scores for all three criteria exceeded 4, the scores for the structure preservation criterion are comparatively lower. Specifically, for conversational-style text such as on Hackernews, the paraphraser LLM occasionally fails to maintain the original structure.

Ablation Studies

Number of Samples

The p-value for any of the datasets we test goes below the threshold after 100-150 samples, suggesting that this is the minimum number of samples needed (Figure 4). The p-value for non-members is always above the threshold.

Ablations on Scoring Model

We investigate the effect of using different scoring models for the ranking of paraphrases. We aim to ascertain whether changing the scoring model would substantially affect the outcomes of SPECTRA. We utilize paraphrases derived from the PeS2o dataset. The OLMo-1b model was originally employed for scoring the PeS2o dataset in our experiments. We compare the rank ordering of paraphrases generated by OLMo-1b against OLMo-7b, Pythia 2.8b, Pythia 160m, and

Method	PubMed			Wiki			HN			PeS2o		
	M	NM	NM/M	M	NM	NM/M	M	NM	NM/M	M	NM	NM/M
LLM-DI	0.06	**0.48**	7.67	0.02	**0.44**	22	0.49	**0.35**	0.71	0.02	**0.17**	8.50
STAMP	0.01	**0.48**	48	0.17	**0.03**	0.19	7E-4	**0.15**	214	0.15	**0.46**	3.07
Maximum	0.03	**1.00**	3.33	1.00	**1.00**	1	**3E-6**	**1.00**	3E5	0.95	**1.00**	1.05
Random	**1E-7**	**8E-4**	8E3	**5E-9**	2E-5	4E3	**4E-27**	**0.10**	3E25	1E-3	**0.11**	100
SPECTRA	**1E-17**	**0.02**	2E15	**4E-19**	**0.02**	5E16	**3E-60**	**0.59**	2E59	**2E-12**	**3E-3**	2E9

Table 2: p-values for different baselines compared against SPECTRA. Bold indicates a statistically significant result for detecting membership under a threshold of $p < 10^{-4}$ for members (M) using Pythia 410m trained on the watermarked data as the target model. Note that the paraphrases for datasets used during continued pretraining (meant for detection) are different for each row and thus result in different target models. Bold also indicates that for non-members (NM), the p-value was above the threshold, indicating that membership was not falsely detected using Pythia 410m as the target model.

	PubMed	Wiki	HN	PeS2o
P-SP	0.88	0.93	0.76	0.93

Table 3: P-SP scores on paraphrasing quality. P-SP scores measure how well the paraphrase preserves the semantic content of the original document.

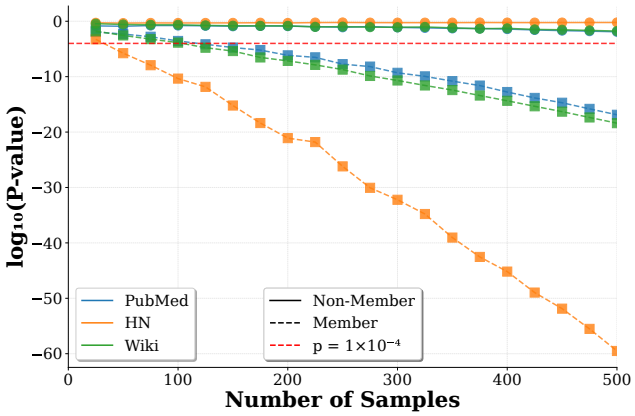

Figure 4: Trend of p-values with number of samples

Pythia 6.9b. Each of these models, along with the original scoring model, is used to compute Min-K%++ scores for the paraphrases from the PeS2o dataset. The rankings generated by each model are then used to compute the Spearman (Spearman 2010) and Kendall (Kendall 1938) rank correlation scores between the original scoring model (OLMo-1b) and the additional scoring models.

For three out of the four additional models, the Spearman's rank correlation coefficient (ρ) values exceed 0.8 (Table 4). The correlation for the Pythia 160m model is lower than that of others. The correlation is above 0.8 for models with more than 2.8b parameters, indicating that the correlation stabilizes for larger models. For Kendall's τ, a value of > 0.6 is considered a strong agreement (Lyu et al. 2021). For three out of four datasets, the τ score > 0.6. These findings suggest that SPECTRA is robust to changing the scoring model as long as the model is sufficiently capable.

Additional Model	Spearman ρ	Kendall τ
Olmo-7b	0.826	0.639
Pythia-2.8b-deduped	0.824	0.635
Pythia-160m-deduped	0.699	0.514
Pythia-6.9b	0.818	0.631

Table 4: Rank-correlation coefficients between OLMo-1b and other scoring models.

Conclusions

We presented SPECTRA, a watermarking approach that enables content creators to test if their data was used to train an LLM. Unlike previous approaches, SPECTRA does not need access to the decoding layer of a large LLM. SPECTRA also does not require access to a held-out dataset from the same domain that is typically necessary for MIA. We empirically show that SPECTRA can achieve a p-value gap of at least nine orders of magnitude between member and non-member data with 500 samples, making this a reliable test of membership.

We highlight some important limitations of our study. We continue to pre-train the Pythia 410m model over a large number of tokens instead of training an LLM from scratch due to computational constraints. We assume access to the log probabilities output by the model, which can be challenging to achieve for proprietary models. Our approach will not help content creators who have already published their content, but can only help them going forward, as the watermarking must be done before publishing. For text data with some structure in it, such as conversational text, our paraphrasing approach does not perform very well.

In terms of future directions, there is a need to develop watermarking techniques that can be used to verify membership not only by the content creator but also by interested third parties. Additional verification of our techniques at the scale of pre-training a model would inspire greater confidence that it can be employed in the event of any legal challenges.

Acknowledgements

This paper was prepared for informational purposes by the Artificial Intelligence Research group of JPMorgan Chase & Co. and its affiliates ("JP Morgan") and is not a product of the Research Department of JP Morgan. JP Morgan makes no representation and warranty whatsoever and disclaims all liability, for the completeness, accuracy, or reliability of the information contained herein. This document is not intended as investment research or investment advice, or a recommendation, offer or solicitation for the purchase or sale of any security, financial instrument, financial product, or service, or to be used in any way for evaluating the merits of participating in any transaction, and shall not constitute a solicitation under any jurisdiction or to any person, if such solicitation under such jurisdiction or to such person would be unlawful.

References

Bouaziz, W.; Videau, M.; Usunier, N.; and El-Mhamdi, E.-M. 2025. Winter Soldier: Backdooring Language Models at Pre-Training with Indirect Data Poisoning. *arXiv preprint arXiv:2506.14913*.

Brittain, B. 2024. Authors sue Anthropic for copyright infringement over AI training. *Reuters*. Technology/Artificial Intelligence section.

Das, D.; Zhang, J.; and Trantèr, F. 2025. Blind baselines beat membership inference attacks for foundation models. In *2025 IEEE Security and Privacy Workshops (SPW)*, 118–125. IEEE.

Duan, M.; Suri, A.; Mireshghallah, N.; Min, S.; Shi, W.; Zettlemoyer, L.; Tsvetkov, Y.; Choi, Y.; Evans, D.; and Hajishirzi, H. 2024. Do Membership Inference Attacks Work on Large Language Models? In *Conference on Language Modeling (COLM)*.

Gao, L.; Biderman, S.; Black, S.; Golding, L.; Hoppe, T.; Foster, C.; Phang, J.; He, H.; Thite, A.; Nabeshima, N.; et al. 2020. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*.

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

Groeneveld, D.; Beltagy, I.; Walsh, E.; Bhagia, A.; Kinney, R.; Tafjord, O.; Jha, A.; Ivison, H.; Magnusson, I.; Wang, Y.; Arora, S.; Atkinson, D.; Authur, R.; Chandu, K.; Cohan, A.; Dumas, J.; Elazar, Y.; Gu, Y.; Hessel, J.; Khot, T.; Merrill, W.; Morrison, J.; Muennighoff, N.; Naik, A.; Nam, C.; Peters, M.; Pyatkin, V.; Ravichander, A.; Schwenk, D.; Shah, S.; Smith, W.; Strubell, E.; Subramani, N.; Wortsman, M.; Dasigi, P.; Lambert, N.; Richardson, K.; Zettlemoyer, L.; Dodge, J.; Lo, K.; Soldaini, L.; Smith, N.; and Hajishirzi, H. 2024. OLMo: Accelerating the Science of Language Models. In Ku, L.-W.; Martins, A.; and Srikumar, V., eds., *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 15789–15809. Bangkok, Thailand: Association for Computational Linguistics.

Huh, M.; Cheung, B.; Wang, T.; and Isola, P. 2024. Position: The platonic representation hypothesis. In *Forty-first International Conference on Machine Learning*.

Kandpal, N.; Lester, B.; Raffel, C.; Majstorovic, S.; Biderman, S.; Abbasi, B.; Soldaini, L.; Shippole, E.; Cooper, A. F.; Skowron, A.; et al. 2025. The Common Pile v0. 1: An 8TB Dataset of Public Domain and Openly Licensed Text. *arXiv preprint arXiv:2506.05209*.

Kendall, M. G. 1938. A new measure of rank correlation. *Biometrika*, 30(1-2): 81–93.

Kirchenbauer, J.; Geiping, J.; Wen, Y.; Shu, M.; Saifullah, K.; Kong, K.; Fernando, K.; Saha, A.; Goldblum, M.; and Goldstein, T. 2023. On the reliability of watermarks for large language models. *arXiv preprint arXiv:2306.04634*.

Krishna, K.; Song, Y.; Karpinska, M.; Wieting, J.; and Iyyer, M. 2023. Paraphrasing evades detectors of ai-generated text, but retrieval is an effective defense. *Advances in Neural Information Processing Systems*, 36: 27469–27500.

Likert, R. 1932. A technique for the measurement of attitudes. *Archives of psychology*.

Lyu, Y.; Rajbahadur, G. K.; Lin, D.; Chen, B.; and Jiang, Z. M. 2021. Towards a consistent interpretation of aiops models. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 31(1): 1–38.

Magnusson, I.; Bhagia, A.; Hofmann, V.; Soldaini, L.; Jha, A. H.; Tafjord, O.; Schwenk, D.; Walsh, E.; Elazar, Y.; Lo, K.; et al. 2024. Paloma: A benchmark for evaluating language model fit. *Advances in Neural Information Processing Systems*, 37: 64338–64376.

Maini, P.; Jia, H.; Papernot, N.; and Dziedzic, A. 2024. LLM Dataset Inference: Did you train on my dataset?

Rastogi, S.; Maini, P.; and Pruthi, D. 2025. STAMP Your Content: Proving Dataset Membership via Watermarked Rephrasings. arXiv:2504.13416.

Rosenblat, S.; O'Reilly, T.; and Strauss, I. 2025. Beyond Public Access in LLM Pre-Training Data. *arXiv preprint arXiv:2505.00020*.

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:2310.16789.

Soldaini, L.; Kinney, R.; Bhagia, A.; Schwenk, D.; Atkinson, D.; Authur, R.; Bogin, B.; Chandu, K.; Dumas, J.; Elazar, Y.; et al. 2024. Dolma: An open corpus of three trillion tokens for language model pretraining research. *arXiv preprint arXiv:2402.00159*.

Spearman, C. 2010. The proof and measurement of association between two things. *International journal of epidemiology*, 39(5): 1137–1150.

Stempel, J. 2023. NY Times sues OpenAI, Microsoft for infringing copyrighted work. *Reuters*. Legal/Transactional section.

Wei, J.; Wang, R.; and Jia, R. 2024. Proving membership in LLM pretraining data via data watermarks. In Ku, L.-W.; Martins, A.; and Srikumar, V., eds., *Findings of the Association for Computational Linguistics: ACL 2024*, 13306–13320. Bangkok, Thailand: Association for Computational Linguistics.

Wieting, J.; Gimpel, K.; Neubig, G.; and Berg-kirkpatrick, T. 2022. Paraphrastic Representations at Scale. In Che, W.; and Shutova, E., eds., *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 379–388. Abu Dhabi, UAE: Association for Computational Linguistics.

Yan, J.; Gupta, V.; and Ren, X. 2022. Textual backdoor attacks with iterative trigger injection. *arXiv preprint arXiv:2205.12700*.

Yeom, S.; Giacomelli, I.; Fredrikson, M.; and Jha, S. 2018. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st computer security foundations symposium (CSF)*, 268–282. IEEE.

Zhang, J.; Sun, J.; Yeats, E.; Ouyang, Y.; Kuo, M.; Zhang, J.; Yang, H. F.; and Li, H. 2025. Min-K%++: Improved Baseline for Pre-Training Data Detection from Large Language Models. In *The Thirteenth International Conference on Learning Representations*.

Zhang, W.; Zhang, R.; Guo, J.; de Rijke, M.; Fan, Y.; and Cheng, X. 2024. Pretraining Data Detection for Large Language Models: A Divergence-based Calibration Method. In Al-Onaizan, Y.; Bansal, M.; and Chen, Y.-N., eds., *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 5263–5274. Miami, Florida, USA: Association for Computational Linguistics.

Zhao, B.; Maini, P.; Boenisch, F.; and Dziedzic, A. 2025. Unlocking Post-hoc Dataset Inference with Synthetic Data. In *Forty-second International Conference on Machine Learning*.