Watershed: A Unified Benchmark for End-to-End Data Provenance Evaluation

Anonymous Authors[1]

Abstract

Data provenance aims to determine whether and how a data source has influenced a downstream LLM. Despite growing interest in data provenance research, current methods tend to specialize on specific settings and suffer from fragmented evaluation standards. To address this, we introduce WATERSHED, a unified benchmark and toolkit for end-to-end provenance evaluation. WATERSHED structures data provenance into stage-wise tests spanning data preparation, LLM training, black-box auditing, and downstream applications such as membership audit, multi-owner source attribution, and unlearning verification. We evaluate existing provenance methods such as watermarking and membership inference attacks on WATERSHED, across a wide range of datasets, model families and attacks. Our results confirm that methods vary in effectiveness across different stages and tasks. By providing a unified framework and exposing these failure modes, WATERSHED establishes a rigorous basis for evaluating data provenance methods.

1. Introduction

Data owners have long faced the risk of their published content being scraped, copied, or reused without consent. LLMs exacerbate this problem: LLM training data consists of mixtures of text, code, and other data collected from heterogeneous sources whose downstream use is difficult to audit (Gao et al., 2020; Mökander et al., 2023; Langlais et al., 2026). This training data may contain unauthorized material from a wide range of data owners (Bommasani et al., 2023), raising data provenance problems (Bommasani et al., 2022): once an LLM is released, how can a data owner determine ❶ whether their data was used in training (e.g. *membership audit*) (Carlini et al., 2021), ❷ whether LLM outputs can be attributed back to them (e.g. *source attribution*) (Lu et al., 2025), or ❸ whether a claimed removal procedure has actually reduced the influence of their data (e.g. *machine unlearning verification*) (Cao & Yang, 2015; Yao et al., 2024)? These questions sit at the heart of dataset governance, attribution, regulatory compliance, and broader LLM accountability (Xu et al., 2024).

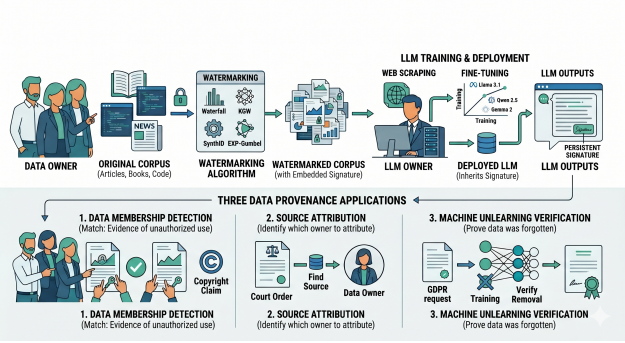

Figure 1. Overview of data-centric provenance. A data owner may release data with an active provenance signal, such as a watermark. If a downstream model is trained on that data, the signal may persist in model outputs and support black-box audits such as membership detection, source attribution, or unlearning verification.

Existing works have addressed these questions about data provenance in LLMs, but largely in isolation. *Membership audit* is typically approached via membership inference attacks (MIAs) (Shokri et al., 2017; Shi et al., 2024; Zhang et al., 2025) that exploit naturally occurring signals of training exposure. *Source attribution* is pursued by actively embedding a data owner's invisible unicode into the training data so that the LLM outputs can be traced back to the correct data owner (Lu et al., 2025). *Machine unlearning verification* is commonly evaluated through verbatim memorization or querying knowledge questions over the forget set (Maini et al., 2024; Shi et al., 2025). These methods use different metrics and are evaluated on different datasets. It is therefore unclear whether a method that succeeds on one question would succeed on another. More importantly, can a single method address all three questions under one shared protocol?

Data-centric watermarking is a natural candidate for this (Lau et al., 2024; Rastogi et al., 2025; Shetty et al., 2026; Lu et al., 2025; 2026). A data owner embeds a unique signature

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

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

into their data before release, and if a downstream model is trained on such data, the same unique signature can be detected by the LLM outputs. This detection process could potentially be used for evaluating all three questions, but there has not been any rigorous experiments done to establish this yet. Existing watermarking benchmarks (Pan et al., 2024; Tu et al., 2024) do not address this gap: they evaluate model-centric watermarking, which targets a distinct problem of identifying LLM-generated text, and hence do not test the requirements that matter for data provenance. The watermark should support multiple data owners, survive adversarial perturbations to the released watermarked dataset, and persist through LLM training.

To address this, we introduce Watershed, a benchmark and toolkit for evaluating data-centric watermarking on an end-to-end audit pipeline for data provenance. Watershed supports stage-wise evaluation of data-level detectability, fidelity, robustness, persistence through training, multi data owner separability, and performance on downstream application tasks such as *membership audit*, *source attribution*, and *machine unlearning verification*. Our key contributions are the following:

- We formulate data provenance as a systemic audit problem spanning data preparation, model training, black-box querying, scoring, and application-level decisions.
- We present Watershed, an extensible benchmark with stage-wise tests for detectability, fidelity, robustness, training persistence, multi-owner separability, membership audit, source attribution, and unlearning verification.
- We systematically evaluate existing methods via Watershed, establishing insights such as showing that high data-level detectability does not guarantee reliable model-level provenance and identifying key failure modes in persistence, robustness, calibration, and attribution.

2. Data-Centric Provenance as Pipeline Evaluation

2.1. Problem Setting and Access Model

Consider N data owners, where owner $i \in \{1, \dots, N\}$ controls a source training dataset corpus $\mathcal{D}_i = \{x_{i,1}, \dots, x_{i,n_i}\}$. An LLM trainer constructs a training corpus $\mathcal{T} = \mathcal{B} \cup \bigcup_{i \in \mathcal{S}} \widetilde{\mathcal{D}}_i$, where $\mathcal{B}$ is background data, $\mathcal{S} \subseteq \{1, \dots, N\}$ is the unknown set of included owners, and $\widetilde{\mathcal{D}}_i$ is the version of owner i's data observed by the trainer. Data owner i applies a watermark $W_{k_i} : \mathcal{D}_i \to \mathcal{D}_i^w$, where k_i is an owner-specific secret key, and the trainer observes $\widetilde{\mathcal{D}}_i = \mathcal{D}_i^w$.

Let M_0 denote a base LLM and let $M_{\mathcal{T}}$ denote the LLM after training or adaptation on $\mathcal{T}$. We focus on black-box auditing. The auditor can query $M_{\mathcal{T}}$ with prompts q and observe responses $y \sim M_{\mathcal{T}}(\cdot \mid q)$, but not the LLM weights, optimizer states, gradients, or training corpus.

Each provenance method defines an owner-specific scoring function $s_i(q, y) \in \mathbb{R}$, that measures evidence that response y to prompt q carries a trace of owner i's data. For watermarking, s_i is a detector score under key k_i.

2.2. Audit Tasks

We evaluate data-centric provenance methods through our three data provenance questions. Each task uses LLM outputs scored for owner-specific evidence, but differs in the decision it must support.

Membership audit. Given a target data owner i, decide whether the audited model was trained on that owner's data (e.g. $H_0 : i \notin \mathcal{S} \quad \text{vs} \quad H_1 : i \in \mathcal{S}$).

For audit prompts $\mathcal{Q}_i$, the auditor aggregates scores $A_i(M_{\mathcal{T}}, \mathcal{Q}_i) := \mathrm{Agg}_{q \in \mathcal{Q}_i,\, y \sim M_{\mathcal{T}}(\cdot|q)}\, s_i(q, y)$ and predicts membership by thresholding $\widehat{m}_i = \mathbf{1}\{A_i(M_{\mathcal{T}}) > \tau_i\}$, where Agg is an aggregation function such as taking the expected value.

Source attribution. Given a candidate set of data owners $\mathcal{C}$ and a batch of prompts $\mathcal{Q}$ whose responses may reflect owner-specific content, identify which data owner's source is most supported by the model outputs. In this task, we compute $\widehat{j} := \arg\max_{j \in \mathcal{C}} A_j(\mathcal{T}, \mathcal{Q}_i)$, where $\mathcal{Q}_i$ refers to queries derived from content in $\mathcal{D}_i$, the data from owner i. A good source attribution method should predict $\widehat{j} = i$.

This is a data owner-level attribution task, not example-level causal attribution. Unlike membership audit, source attribution inherently involve multiple data owners: a method must not only detect that some provenance signal exists, but distinguish among different data owners' signals.

Unlearning verification. Suppose a subset of data owners $\mathcal{F} \subseteq \mathcal{S}$ has requested removal of their data, splitting them from the remaining owners $\mathcal{R}$. The aggregate forget set is $\mathcal{D}^{\mathrm{F}} = \bigcup_{i \in \mathcal{F}} \mathcal{D}_i$ and the corresponding retain set $\mathcal{D}^{\mathrm{R}}$ consists of all training data not associated with $\mathcal{F}$. An unlearning procedure U is applied to the trained model to produce a model $M_{-\mathcal{F}} = U(M_{\mathcal{T}}, \mathcal{D}^{\mathrm{F}})$, such that $M_{-\mathcal{F}}(\cdot|q) \approx M_{\mathcal{R}}(\cdot|q)$ for all queries, where $M_{\mathcal{R}}$ is the model retrained on the retain set $\mathcal{D}^{\mathrm{R}}$. Unlearning verification asks whether the influence of $\mathcal{D}^{\mathrm{F}}$ has been reduced to the level expected under suitable controls. For each forget owner $f \in \mathcal{F}$ we measure the reduction in forget-owner signal $\Delta_f = A_f(M_{\mathcal{T}}, \mathcal{Q}_f) - A_f(M_{-\mathcal{F}}, \mathcal{Q}_f)$ and for retain owner $r \in \mathcal{R}$, we measure the reference retain-owner signal $R_r = A_r(M_{-\mathcal{F}}, \mathcal{Q}_r) - A_r(M_{\mathrm{R}}, \mathcal{Q}_r)$.

3. Evaluation Criteria and Metrics

The effectiveness of watermarking as a data provenance method on the audit tasks depends on its reliability across various stages and failure points. A provenance method can fail at multiple points before an audit decision is made. For example, the provenance signal may already be weak in the prepared data, degrade data quality, may be removed by pre-processing (such as paraphrasing), may fail to persist through training, may collide with another owner's signal, or may be poorly calibrated for the target application. We therefore evaluate provenance methods through stage-wise failure points.

Data-level verifiability. Given watermarked data $\mathcal{D}_i^w$, the detector under the correct key should distinguish it from unwatermarked and wrong-key data:

$$s_i(x) > s_j(x) \quad \text{for } x \in \mathcal{D}_i^w,\ j \neq i.$$

This stage verifies the watermarking operation itself. It is necessary, but not sufficient, for LLM-level provenance.

Fidelity. A watermark should preserve the fidelity of the source data. If x is an original text and x^w its watermarked counterpart, x^w should preserve the meaning, fluency, and downstream utility of x. We evaluate this using semantic similarity, fluency, and task utility where applicable. Fidelity matters since a provenance mechanism that makes the data unnatural, unusable, or distributionally distorted may be detectable but impractical.

Robustness to transformations. The signal should survive realistic transformations of either the data or the observed outputs. Let g denote a perturbation such as token deletion, synonym substitution, or paraphrasing. A robust signal should remain detectable after transformation, i.e., $s_i(g(x)) > \tau_i$, while maintaining low false-positive rates on non-member and wrong-key controls. Paraphrasing is especially important because it can remove surface signals while preserving semantic content (data fidelity).

Persistence through training. The defining requirement of data-centric provenance is that the signal survive model training. If $M_{\mathcal{T}}$ is trained on owner i's prepared data, then responses from $M_{\mathcal{T}}$ should exhibit stronger owner-specific evidence than responses from a reference model (e.g. $A_i(M_{\mathcal{T}}, \mathcal{Q}_i) > A_i(M_{\text{ref}}, \mathcal{Q}_i)$ that has never been trained on $\mathcal{D}_i$. This is the main distinction between data-centric provenance and ordinary text-level watermark detection. Training is a lossy transformation since it mixes the owner's data with background data, compresses the signal into model parameters, and exposes it only indirectly through generated outputs.

Multi-owner separability. In realistic settings, many data owners may use the same provenance mechanism (e.g. watermarking family). The correct owner score should therefore exceed scores under other owners' keys:

$$A_i(M_{\mathcal{T}}, \mathcal{Q}_i) > A_j(M_{\mathcal{T}}, \mathcal{Q}_i) \qquad \text{for } j \neq i.$$

This criterion is necessary for source attribution and for avoiding collisions among owners. It also distinguishes owner-specific provenance from generic distribution shifts that merely indicate that some training exposure occurred.

Application validity. Finally, the signal must support the intended audit decision. *Membership audit* requires calibrated discrimination between member and non-member owners. *Source attribution* requires accurate ranking among candidate owners. *Machine unlearning verification* requires measuring residual target data owner signal relative to controls. These application-level requirements are related but not interchangeable, and methods effective for one might not be for the others.

4. Watershed: A Benchmark for Data-Centric Provenance

The previous section defines data-centric provenance as a systemic, multi-stage evaluation problem. Watershed operationalizes this view in a benchmark and toolkit for evaluating provenance methods under shared assumptions. Rather than treating watermark detection, *membership inference*, *source attribution*, and *machine unlearning verification* as unrelated tasks, Watershed exposes them as different audit decisions over a common pipeline: construct owner-specific data, apply the watermark, train an LLM, query the LLM, score responses under candidate data owners, and evaluate the resulting audit decision.

Watershed is designed to answer two broad questions. First, *where in the pipeline does a provenance method succeed or fail?* Second, *how does watermarking compare against baseline methods when evaluated under matched access LLMs, datasets, and controls?* This section describes the benchmark design, components, and evaluation.

4.1. Benchmark Overview

Given a collection of owner corpora $\{\mathcal{D}_i\}_{i=1}^N$, Watershed proceeds through five stages, with attacks and evaluations applied across the stages:

1. Owner construction. The benchmark begins by partitioning a dataset into owner-specific corpora. Data owners may correspond to natural sources, such as authors, publishers, domains, or programming tasks, or to synthetic partitions constructed for controlled experiments.

2. Provenance preparation. Each data owner is assigned a key k_i, and the training subset is transformed into a watermarked corpus $\mathcal{D}_i^w = W_{k_i}(\mathcal{D}_i^{\text{train}})$.

3. LLM training. A base LLM M_0 is trained or adapted

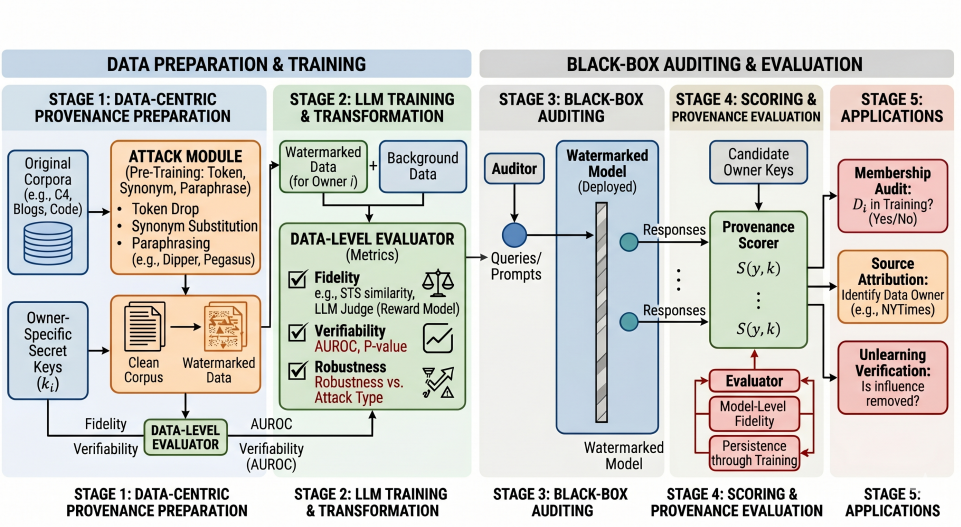

Figure 2. Overview of Watershed

on a mixture of owner data and background data $\mathcal{T} = \mathcal{B} \cup \bigcup_{i \in \mathcal{S}} \widetilde{\mathcal{D}}_i$, where $\mathcal{S}$ is the set of included owners and $\widetilde{\mathcal{D}}_i$ is the watermarked version of owner i's data. This stage converts a data-level provenance signal into a LLM-level audit problem.

4. Black-box audit. The trained model is queried using prompts from held-out audit data. For each prompt q, the model produces a response $y \sim M_{\mathcal{T}}(\cdot \mid q)$. The benchmark then scores the response under one or more candidate owners using method-specific scoring functions $s_i(q, y)$.

5. Application evaluation. Finally, Watershed aggregates scores and evaluates the corresponding audit decision: membership, source attribution, or unlearning verification. The same generated responses can therefore be analyzed at multiple levels: as evidence of training exposure, as evidence for a particular owner, or as residual evidence after a removal procedure.

4.2. Components

Watershed consists of modular components for data preparation, provenance methods, model training, auditing, and evaluation. This modularity is important because existing data-centric provenance methods are often evaluated under bespoke pipelines, making direct comparison difficult. Prior methods are proposed and evaluated in isolation, while existing watermarking benchmarks focus on model-centric watermarking rather than data-centric provenance tasks.

Datasets. The benchmark supports datasets that can be partitioned into source-level owners. In our experiments, we use text and code corpora spanning news, blogs, and programming tasks. These datasets allow us to evaluate provenance across different content types and owner structures. Natural owner groupings are used when available; otherwise, we construct synthetic owners by sampling disjoint subsets. In our current implementation, we considered: C4-realnewslike for news articles (Raffel et al., 2020), Blog Authorship Corpus for blogs (Schler et al., 2006), and MBPP/MBJSP for code (Austin et al., 2021).

Watermarks. We use watermarking algorithms for our data provenance setting. Each watermarking algorithm provides two functions: to apply a watermark $W_{k_i}(\cdot)$ that prepares data owner i's corpus, and a scoring function $s_i(\cdot)$ that measures the watermark's signal under key k_i. For the benchmark, we apply the data-centric watermarking framework proposed by (Lau et al., 2024) that enables other model-centric LLM watermarking methods to be adapted to the data provenance setting. In Watershed, we include Waterfall (Lau et al., 2024), KGW (Kirchenbauer et al., 2023), Unigram (Zhao et al., 2024), EXP-Gumbel (Aaronson, 2022), SynthID (Dathathri et al., 2024).

Baselines. We include MIAs for *membership audit*. These methods do not modify the training data. Instead, they score naturally occurring exposure signals, such as likelihood, perplexity, or reference-model differences. Their role is not to solve all provenance tasks, but to establish how much membership evidence is available without active intervention (such as watermarking).

Models. The benchmark supports multiple open LLM families. This is important because watermark signals may depend on tokenizer, model distribution, training dynamics, and generation behavior. We therefore distinguish matched settings, where the LLM used to prepare data and the LLM trained on that data come from the same family, from cross-family settings, where they differ. In our implementation, we considered the models: Llama 3.1 8B (Grattafiori et al.,

2024) and Gemma 2 9B (Team et al., 2024).

Attacks. Watershed includes data-side and output-side perturbations for testing robustness. Data-side perturbations include token deletion, synonym substitution, and paraphrasing. Output-side perturbations can be used to test whether audit scores remain stable when generated responses are edited or rewritten. In the main experiments, we focus on data-side robustness and training persistence, treating model training itself as the central lossy transformation.

4.3. Evaluation Stages

The benchmark reports both stage-wise diagnostics and application-level outcomes. This separation is central to Watershed: when a method fails on an audit task, stage-wise diagnostics help identify whether the failure comes from weak data-level signal, poor fidelity, lack of robustness, loss during training, cross-owner interference, or poor calibration.

Data-stage evaluation. Before training any LLM, Watershed evaluates the prepared corpus directly. This includes data-level detectability, wrong-key separability, robustness to corpus perturbations, and fidelity with respect to the original data. These tests answer whether the watermark signal was successfully embedded and whether the watermarked data remains useful.

Model-stage evaluation. After training, the benchmark evaluates whether the signal persists in LLM outputs. The trained LLM is queried using held-out prompts, and generated responses are scored under candidate data owners. The key comparison is between the trained model and suitable controls, such as a base LLM, a clean fine-tuned LLM, or wrong-owner keys. This stage is the defining test for data provenance: a signal that does not survive training cannot support downstream LLM audit.

Application-stage evaluation. Finally, Watershed evaluates whether the scores support the intended audit decision. For *membership audit*, we report discrimination between member and non-member data owners. For *source attribution*, we report whether the correct data owner can be attributed based on the LLM outputs. For unlearning verification, we report whether the LLM still contains influence of the data.

5. Empirical Analysis of Data-centric Watermarking

For the data owner to confidently claim that the LLM has been trained on their data, we must first perform rigorous evaluations on the effectiveness of watermarks in the data-centric setting. We split our analysis on the data-side (Section 5.2) and model-side (Section 5.3). This ensures that we can properly verify whether the uploaded data contains the watermark signal, and whether the LLM trained on the watermarked data generates outputs that contains the same watermark signal.

5.1. Experimental Setup

We focus the main paper on the C4 realnewslike dataset since most LLM data-provenance disputes to date involve news publishers. We partition the corpus into 10 data owners with 1,000 articles each, watermarked under owner-specific keys. The base LLMs are fine-tuned for 3 epochs at a learning rate of `2e-5`. For each watermarking scheme we use the default hyperparameters from its original paper, measuring out-of-the-box effectiveness in the data-centric setting. Audit-time detection queries the LLM with $K = 10$ candidate keys (one per data owner) using a 50-token prefix, and we report AUROC as the threshold-free metric and TPR at FPR levels of 1%, 5%, and 10%. Full settings appear in App D; results on the Blog Authorship Corpus are in App F.1 and MBPP/MBJSP datasets are in App F.2.

5.2. Effectiveness of Watermark on Corpus

5.2.1. Detection and Robustness to Attacks

We first evaluate the verifiability of the watermarks before model training. Table 1 reports AUROC of the detector on watermarked vs unwatermarked text. It shows that each watermark embeds a strong signal into the dataset, shown by the relatively high AUROC scores before attacks. Note that while attacks such as token drop and synonym substitution may reduce verifiability slightly, paraphrase attacks can effectively remove watermarks for several watermarking schemes (details in App D.2.3). However, Waterfall and KGW remains relatively robust to all attacks.

5.2.2. Does Watermarking Degrade Text Quality?

Data owners may be concerned about the utility tradeoff when using watermarks, so maintaining fidelity is an important property of a good watermark. To evaluate the fidelity of the text we use the STS scores and an LLM Judge (Gu et al., 2025) (details in App D.1).

In Table 13, we notice that the STS scores between the watermarked and original text is very high at around 0.9 and the LLM Judge only identifies $< 3\%$ of the watermarked text as unfaithful to the original text which suggests that the watermark only slightly deviates from the original text.

5.2.3. Distinguishing Between Multiple Data Owners

As per the multi-owner separability property, we also expect to be able to distinguish between the N watermark

Watermark	Model	Before Attack	After Attack		
			Tok. Drop	Syn. Sub.	Paraphrase
Waterfall	Llama-8B	**1.0000**	**1.0000**	**1.0000**	**0.9800**
	Gemma-9B	**1.0000**	**1.0000**	**1.0000**	0.9375
KGW	Llama-8B	**1.0000**	**1.0000**	**1.0000**	0.9390
	Gemma-9B	**1.0000**	0.9987	0.9987	0.9145
Unigram	Llama-8B	0.9828	0.9700	0.9657	0.8435
	Gemma-9B	0.9748	0.9518	0.9465	0.8105
SynthID	Llama-8B	0.9973	0.9857	0.9908	0.7705
	Gemma-9B	0.9949	0.9462	0.9654	0.6725
EXP-Gumbel	Llama-8B	0.9994	0.9969	0.9970	0.8490
	Gemma-9B	0.9943	0.9737	0.9734	0.7595

Table 1. AUROC scores on the C4 realnewslike dataset before and after attacks. Paraphrase is using Llama 3.1 8B.

keys. The watermark's per-key signal should not collide across data owners as the detector instantiated with key k_i must distinguish between data from $\mathcal{D}_i$ and data $\mathcal{D}_j$ where $j \neq i$ at the data level, before any model is trained on the watermarked corpus.

Table 11 shows that the per-key AUROC ≈ 1 across our various configurations, where AUROC discriminates rows from $\mathcal{D}_i^w$ scored under key k_i (positives) from rows under any other key $k_j, j \neq i$ (negatives). This empirically shows there is no cross-key interference at the data level as the keys uniquely identify their data owner's content and this establishes the precondition for the model-side attribution claim.

Watermark	Model	AUROC
Waterfall	Llama-8B	0.942
	Gemma-9B	**0.991**
KGW	Llama-8B	0.880
	Gemma-9B	**0.987**
Unigram	Llama-8B	0.858
	Gemma-9B	**0.971**
SynthID	Llama-8B	0.615
	Gemma-9B	0.921
EXP-Gumbel	Llama-8B	0.762
	Gemma-9B	**0.957**

Table 2. Model-side detection AUROC on C4-realnewslike with K=10 candidate keys.

5.3. Watermark Persistence through Training

After the data owner has verified that the watermark signal is strong in the watermarked dataset, and the fidelity of the text is high, we investigate next the influence of the watermark on the LLM trained on the watermarked datasets.

5.3.1. Can watermarks persist through training?

An important property for data-centric watermarking is the ability for the signal to persist through training. Concretely, if a model $M_{\theta'}$ is fine-tuned on $D_i^W = \mathcal{W}_{k^{(i)}}(\mathcal{D}_i)$, its outputs $z \sim M_{\theta'}$ should still satisfy $s(z; k^{(i)}) > \tau$ when prompted with reasonable queries.

To test this, we fine-tune each base LLM on its own watermarked corpus (e.g. if the corpus was watermarked using Waterfall applying Llama-8B as paraphraser, we use this watermarked corpus to fine-tune Llama-8B) and evaluate the performance of the watermark detector. We compute the AUROC to distinguish between the outputs of an LLM trained on watermarked text vs non-watermarked text.

Table 2 indicates signs that the watermarks persist through training across all five watermark families: every scheme exceeds the random-guess baseline of 0.5, and the distributional schemes (Waterfall, KGW, Unigram) cluster above 0.86 on Llama-8B and above 0.97 on Gemma-9B. The non-distortionary schemes (SynthID, EXP-Gumbel) retain a weaker but still present signal on Llama-8B (0.62 and 0.76 respectively) and recover strongly on Gemma-9B (0.92 and 0.96). Treating fine-tuning as a lossy transformation analogous to paraphrase or compression, the watermark is preserved through this transformation.

We compare this against using MIA attacks on an unwatermarked setting. In this setting, the LLM is fine-tuned on the original corpus (without the watermark). The data owner may check whether their data has been used by using MIA methods, and we note that the AUROC is high at ≈ 0.98 (see Table 20).

However, we must also consider the robustness to attacks. As we have established, it is highly likely that the published data may undergo perturbations before training. Hence, we must also consider whether provenance methods still hold if

Direction	KGW	Unigram	Waterfall	SynthID	EXP-Gumbel
LlamaGen → GemmaTrain	0.979	0.925	**0.992**	0.897	0.972
GemmaGen → LlamaTrain	0.781	0.823	**0.856**	0.602	0.729

Table 3. Cross-family multi-query model-side detection AUROC on C4-realnewslike with $K=10$ candidate keys. Watermark detector uses the generator-side tokenizer.

the data was attacked before training. We use the paraphrase attack since it is the most effective perturbation compared to token drop and synonym substitution.

In Table 4, the watermark column reveals the ability of the detector to distinguish between the watermarked vs unwatermarked LLM outputs and Min-K%++ measures seen training rows vs. an unseen holdout. In this setting, the trained LLM never saw the watermarked or original text directly — only paraphrases of the watermark.

		Watermark	MIA
Watermark	**Model**	**AUROC**	**Min-K%++**
Waterfall	Llama-8B	0.814	**0.488**
	Gemma-9B	**0.837**	0.475
KGW	Llama-8B	0.799	0.487
	Gemma-9B	0.814	0.471
Unigram	Llama-8B	0.725	0.482
	Gemma-9B	0.821	0.460
SynthID	Llama-8B	0.576	0.477
	Gemma-9B	0.654	0.445
EXP-Gumbel	Llama-8B	0.654	0.486
	Gemma-9B	0.673	0.443

Table 4. The trained model never saw the watermarked or original text directly — only paraphrases of the watermarked text. Watermarks are still robust to paraphrasing while Min-K%++ goes down to random chance.

The results show that data-centric watermarking outperforms Min-K%++ (Zhang et al., 2025) by a significant margin when we consider an attack on the training data.

5.3.2. Different Model Families for Watermarker vs Trainer

Realistic deployments mismatch the watermark generator and trainer families. Table 3 reports two cross-family directions, where `XGen` → `YTrain` denotes that LLM X is used as the watermark generator and LLM Y is fine-tuned on the resulting watermarked corpus. Every watermark persists through the cross-family trainer, and Waterfall under `LlamaGen` → `GemmaTrain` reaches 0.992 — *higher* than the same-family Gemma→Gemma configuration, since the base LLM from a different family produces output far from the watermark distribution and widens the finetuned-vs-base gap. The reverse `GemmaGen` → `LlamaTrain` (0.856) is lower, so the cross-family gain is asymmetric and depends on the trainer absorbing the watermark distinctly from its natural outputs.

5.3.3. Impact of Watermark on Model Utility

We also investigate the impact of the watermarks on model utility, given in Table 21. We test our LLMs trained on watermarks based on the average zero-shot accuracy across six LM Evaluation Harness (Gao et al., 2024) tasks. The results show that training the LLM on watermarked data does not significantly impact model utility on general tasks. In fact, the degradation of fine-tuning on a watermarked dataset is not too different from the degradation suffered from fine-tuning on a non-watermarked dataset.

5.3.4. Source Attribution

In the event the data owner successfully verifies that the LLM was trained on their data, one possible resolution is for the LLM owner to compensate the data owner with royalty payments. Rather than attributing each individual output to a single source, we measure the aggregate per-source signal across an evaluation batch and use it as the basis for proportional royalty distribution at periodic settlement. We pool by source: per-source mean score across an evaluation batch, argmax over candidates (more details in App C.2). Pooled accuracy reaches 1.0 for every (watermark, model) pair except Unigram-Llama (0.8); see Table 12.

5.3.5. Machine Unlearning Verification

An alternative solution that the court might rule is that the LLM owner must remove the influence of the data owner's content, which is also known as *machine unlearning* (Cao & Yang, 2015; Yao et al., 2024).

It is important to be able to evaluate whether the influence of the data is still present in the LLM, which is proper machine unlearning verification metrics (Shi et al., 2025; Maini et al., 2024). Recently, there has been work on utilizing watermarks for such a task which shows to be promising (Lu et al., 2026). For unlearning tasks, the "gold standard" is the LLM trained on only the retain set. Ideally, the performance on the forget set should be low while the performance on the retain set is high. The AUROC follows the same model-side convention as Section 5.3 but evaluated separately on prompts from $\mathcal{D}^F$ (forget) and $\mathcal{D}^R$ (retain).

On the forget side, gradient difference reduces AUROC to-

Watermark	Model	Forget Set AUROC ↓ original	GD	oracle
KGW	Llama-8B	0.616	0.481	0.477
	Gemma-9B	0.655	**0.489**	0.464
Unigram	Llama-8B	0.734	**0.471**	0.513
	Gemma-9B	0.705	**0.639**	0.510
Waterfall	Llama-8B	0.654	0.448	0.490
	Gemma-9B	0.553	**0.569**	0.427
SynthID	Llama-8B	0.524	**0.492**	0.568
	Gemma-9B	0.439	**0.422**	0.437
EXP-Gumbel	Llama-8B	0.497	**0.437**	0.467
	Gemma-9B	0.526	**0.440**	0.451

Watermark	Model	Retain Set AUROC ↑ original	GD	oracle
KGW	Llama-8B	0.745	0.652	0.753
	Gemma-9B	0.729	0.691	0.729
Unigram	Llama-8B	0.747	0.662	0.753
	Gemma-9B	0.697	0.677	0.682
Waterfall	Llama-8B	0.805	0.722	0.807
	Gemma-9B	0.708	0.680	0.731
SynthID	Llama-8B	0.512	0.521	0.494
	Gemma-9B	0.531	0.515	0.522
EXP-Gumbel	Llama-8B	0.570	0.509	0.594
	Gemma-9B	0.570	0.547	0.592

Table 5. Unlearning results (AUROC) across watermarks and base models. *Top*: forget set (lower is better; close to oracle = effective unlearning). *Bottom*: retain set (higher is better; close to oracle = utility preserved). `original` = LLM fine-tuned on full dataset; `GD` = gradient difference; `oracle` = retrain-on-retain.

ward the oracle for KGW (Llama: $0.616 \rightarrow 0.481$ vs. oracle 0.477; Gemma: $0.655 \rightarrow 0.489$ vs. 0.464), but overshoots below the oracle on Unigram-Llama and Waterfall-Llama or barely moves on Waterfall-Gemma. On the retain side, GD costs 0.07–0.10 AUROC vs. oracle on Llama-8B across watermarks — a measurable utility drop — while staying close to oracle on Gemma-9B. SynthID sits at the random-guess baseline on both sides. Reading both sides together exposes whether an operator is removing the right amount of signal from the forget set and retaining enough performance on the retain set.

6. Related Works

6.1. Data-centric Watermarking

Data-centric watermarking embeds a watermark directly into a training corpus rather than into model outputs, so that any model trained on the corpus inherits a detectable signal. Existing methods pursue distinct goals: imperceptible embedding via invisible Unicode characters (Lu et al., 2025), paraphrase-based rewriting that wraps a model-centric watermark to produce token-biased training text (Lau et al., 2024), and watermark persistence as evidence of incomplete machine unlearning (Lu et al., 2026). Closely related is the line of work on *radioactivity* (Sander et al., 2024), which characterizes when the signal of an output watermark persists into a downstream model trained on watermarked text. These methods have been evaluated under bespoke protocols on different model families and datasets, and to our knowledge have never been directly compared. Watershed addresses this gap with a shared interface and a unified evaluation protocol.

6.2. Watermarking Benchmarks and Toolkits

Two prior efforts establish benchmarks or toolkits for LLM watermarking. WaterBench (Tu et al., 2024) benchmarks watermarks across four generation tasks under a matched-strength comparison protocol, while MarkLLM (Pan et al., 2024) provides reference implementations of watermarks behind a unified API. Both target the *model-centric* regime: they evaluate watermarks on perturbations of generated text from a watermarked sampler, not on generations from a model trained on watermarked data. Neither can therefore assess *membership audit*, *source attribution*, or *machine unlearning verification* in the data-centric sense, since each requires training a model on watermarked data and probing its generations. Watershed is complementary: it targets the data-centric regime, integrates with MarkLLM as an extension bridge for additional algorithms, and is the first benchmark we are aware of that evaluates whether a watermark survives training.

7. Conclusion

We presented Watershed, a unified benchmark and toolkit for evaluating data-centric watermarking as an end-to-end provenance audit pipeline. Watershed structures provenance evaluation into stage-wise tests for data-level detectability, fidelity, robustness, training persistence, and multi-owner separability, together with three downstream audit tasks: membership audit, source attribution, and unlearning verification. Our empirical study across five watermarking schemes, two LLM families, and three data domains shows that data-centric watermarking can support all three audit tasks under a single shared protocol, but a watermark's ranking on one pipeline stage does not predict its ranking on another: SynthID is essentially perfect at the data level but loses substantial signal during training; Unigram is robust through training but yields the weakest attribution; and when a watermark's model-side signal is already near chance (e.g., Waterfall-Gemma, SynthID-Llama), the unlearning metric itself loses resolution. These cross-stage gaps justify the stage-wise design — no single number captures whether a

watermark is fit-for-purpose across the full audit pipeline.

References

Aaronson, S. My ai safety lecture for UT effective altruism. Shtetl-Optimized (Blog), November 2022. URL https://scottaaronson.blog/?p=6823.

Austin, J., Odena, A., Nye, M., Bosma, M., Michalewski, H., Dohan, D., Jiang, E., Cai, C., Terry, M., Le, Q., and Sutton, C. Program synthesis with large language models, 2021. URL https://arxiv.org/abs/2108.07732.

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N., Chen, A., Creel, K., Davis, J. Q., Demszky, D., Donahue, C., Doumbouya, M., Durmus, E., Ermon, S., Etchemendy, J., Ethayarajh, K., Fei-Fei, L., Finn, C., Gale, T., Gillespie, L., Goel, K., Goodman, N., Grossman, S., Guha, N., Hashimoto, T., Henderson, P., Hewitt, J., Ho, D. E., Hong, J., Hsu, K., Huang, J., Icard, T., Jain, S., Jurafsky, D., Kalluri, P., Karamcheti, S., Keeling, G., Khani, F., Khattab, O., Koh, P. W., Krass, M., Krishna, R., Kuditipudi, R., Kumar, A., Ladhak, F., Lee, M., Lee, T., Leskovec, J., Levent, I., Li, X. L., Li, X., Ma, T., Malik, A., Manning, C. D., Mirchandani, S., Mitchell, E., Munyikwa, Z., Nair, S., Narayan, A., Narayanan, D., Newman, B., Nie, A., Niebles, J. C., Nilforoshan, H., Nyarko, J., Ogut, G., Orr, L., Papadimitriou, I., Park, J. S., Piech, C., Portelance, E., Potts, C., Raghunathan, A., Reich, R., Ren, H., Rong, F., Roohani, Y., Ruiz, C., Ryan, J., Ré, C., Sadigh, D., Sagawa, S., Santhanam, K., Shih, A., Srinivasan, K., Tamkin, A., Taori, R., Thomas, A. W., Tramèr, F., Wang, R. E., Wang, W., Wu, B., Wu, J., Wu, Y., Xie, S. M., Yasunaga, M., You, J., Zaharia, M., Zhang, M., Zhang, T., Zhang, X., Zhang, Y., Zheng, L., Zhou, K., and Liang, P. On the opportunities and risks of foundation models, 2022. URL https://arxiv.org/abs/2108.07258.

Bommasani, R., Klyman, K., Longpre, S., Kapoor, S., Maslej, N., Xiong, B., Zhang, D., and Liang, P. The foundation model transparency index, 2023. URL https://arxiv.org/abs/2310.12941.

Cao, Y. and Yang, J. Towards making systems forget with machine unlearning. In *Proceedings of the 2015 IEEE Symposium on Security and Privacy*, SP '15, pp. 463–480, USA, 2015. IEEE Computer Society. ISBN 9781467369497. doi: 10.1109/SP.2015.35. URL https://doi.org/10.1109/SP.2015.35.

Carlini, N., Tramer, F., Wallace, E., Jagielski, M., Herbert-Voss, A., Lee, K., Roberts, A., Brown, T., Song, D., Erlingsson, U., Oprea, A., and Raffel, C. Extracting training data from large language models, 2021. URL https://arxiv.org/abs/2012.07805.

Dathathri, S., See, A., Ghaisas, S., Huang, P.-S., McAdam, R., Welbl, J., Bachani, V., Kaskasoli, A., Stanforth, R., Matejovicova, T., Hayes, J., Vyas, N., Merey, M. A., Brown-Cohen, J., Bunel, R., Balle, B., Cemgil, T., Ahmed, Z., Stacpoole, K., Shumailov, I., Baetu, C., Gowal, S., Hassabis, D., and Kohli, P. Scalable watermarking for identifying large language model outputs. *Nature*, 634(8035):818–823, October 2024. ISSN 1476-4687. doi: 10.1038/s41586-024-08025-4. URL http://dx.doi.org/10.1038/s41586-024-08025-4.

Gao, L., Biderman, S., Black, S., Golding, L., Hoppe, T., Foster, C., Phang, J., He, H., Thite, A., Nabeshima, N., Presser, S., and Leahy, C. The pile: An 800gb dataset of diverse text for language modeling, 2020. URL https://arxiv.org/abs/2101.00027.

Gao, L., Tow, J., Abbasi, B., Biderman, S., Black, S., DiPofi, A., Foster, C., Golding, L., Hsu, J., Le Noac'h, A., Li, H., McDonell, K., Muennighoff, N., Ociepa, C., Phang, J., Reynolds, L., Schoelkopf, H., Skowron, A., Sutawika, L., Tang, E., Thite, A., Wang, B., Wang, K., and Zou, A. The language model evaluation harness, 07 2024. URL https://zenodo.org/records/12608602.

Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Vaughan, A., Yang, A., Fan, A., Goyal, A., Hartshorn, A., Yang, A., Mitra, A., Sravankumar, A., Korenev, A., Hinsvark, A., Rao, A., Zhang, A., Rodriguez, A., Gregerson, A., Spataru, A., Roziere, B., Biron, B., Tang, B., Chern, B., Caucheteux, C., Nayak, C., Bi, C., Marra, C., McConnell, C., Keller, C., Touret, C., Wu, C., Wong, C., Ferrer, C. C., Nikolaidis, C., Allonsius, D., Song, D., Pintz, D., Livshits, D., Wyatt, D., Esiobu, D., Choudhary, D., Mahajan, D., Garcia-Olano, D., Perino, D., Hupkes, D., Lakomkin, E., AlBadawy, E., Lobanova, E., Dinan, E., Smith, E. M., Radenovic, F., Guzmán, F., Zhang, F., Synnaeve, G., Lee, G., Anderson, G. L., Thattai, G., Nail, G., Mialon, G., Pang, G., Cucurell, G., Nguyen, H., Korevaar, H., Xu, H., Touvron, H., Zarov, I., Ibarra, I. A., Kloumann, I., Misra, I., Evtimov, I., Zhang, J., Copet, J., Lee, J., Geffert, J., Vranes, J., Park, J., Mahadeokar, J., Shah, J., van der Linde, J., Billock, J., Hong, J., Lee, J., Fu, J., Chi, J., Huang, J., Liu, J., Wang, J., Yu, J., Bitton, J., Spisak, J., Park, J., Rocca, J., Johnstun, J., Saxe, J., Jia, J., Alwala, K. V., Prasad, K., Upasani, K., Plawiak, K., Li, K., Heafield, K., Stone, K., El-Arini, K., Iyer, K., Malik, K., Chiu, K., Bhalla, K., Lakhotia, K., Rantala-Yeary, L., van der Maaten, L., Chen, L., Tan, L., Jenkins, L., Martin, L., Madaan, L., Malo, L., Blecher, L., Landzaat,

L., de Oliveira, L., Muzzi, M., Pasupuleti, M., Singh, M., Paluri, M., Kardas, M., Tsimpoukelli, M., Oldham, M., Rita, M., Pavlova, M., Kambadur, M., Lewis, M., Si, M., Singh, M. K., Hassan, M., Goyal, N., Torabi, N., Bashlykov, N., Bogoychev, N., Chatterji, N., Zhang, N., Duchenne, O., Çelebi, O., Alrassy, P., Zhang, P., Li, P., Vasic, P., Weng, P., Bhargava, P., Dubal, P., Krishnan, P., Koura, P. S., Xu, P., He, Q., Dong, Q., Srinivasan, R., Ganapathy, R., Calderer, R., Cabral, R. S., Stojnic, R., Raileanu, R., Maheswari, R., Girdhar, R., Patel, R., Sauvestre, R., Polidoro, R., Sumbaly, R., Taylor, R., Silva, R., Hou, R., Wang, R., Hosseini, S., Chennabasappa, S., Singh, S., Bell, S., Kim, S. S., Edunov, S., Nie, S., Narang, S., Raparthy, S., Shen, S., Wan, S., Bhosale, S., Zhang, S., Vandenhende, S., Batra, S., Whitman, S., Sootla, S., Collot, S., Gururangan, S., Borodinsky, S., Herman, T., Fowler, T., Sheasha, T., Georgiou, T., Scialom, T., Speckbacher, T., Mihaylov, T., Xiao, T., Karn, U., Goswami, V., Gupta, V., Ramanathan, V., Kerkez, V., Gonguet, V., Do, V., Vogeti, V., Albiero, V., Petrovic, V., Chu, W., Xiong, W., Fu, W., Meers, W., Martinet, X., Wang, X., Wang, X., Tan, X. E., Xia, X., Xie, X., Jia, X., Wang, X., Goldschlag, Y., Gaur, Y., Babaei, Y., Wen, Y., Song, Y., Zhang, Y., Li, Y., Mao, Y., Coudert, Z. D., Yan, Z., Chen, Z., Papakipos, Z., Singh, A., Srivastava, A., Jain, A., Kelsey, A., Shajnfeld, A., Gangidi, A., Victoria, A., Goldstand, A., Menon, A., Sharma, A., Boesenberg, A., Baevski, A., Feinstein, A., Kallet, A., Sangani, A., Teo, A., Yunus, A., Lupu, A., Alvarado, A., Caples, A., Gu, A., Ho, A., Poulton, A., Ryan, A., Ramchandani, A., Dong, A., Franco, A., Goyal, A., Saraf, A., Chowdhury, A., Gabriel, A., Bharambe, A., Eisenman, A., Yazdan, A., James, B., Maurer, B., Leonhardi, B., Huang, B., Loyd, B., Paola, B. D., Paranjape, B., Liu, B., Wu, B., Ni, B., Hancock, B., Wasti, B., Spence, B., Stojkovic, B., Gamido, B., Montalvo, B., Parker, C., Burton, C., Mejia, C., Liu, C., Wang, C., Kim, C., Zhou, C., Hu, C., Chu, C.-H., Cai, C., Tindal, C., Feichtenhofer, C., Gao, C., Civin, D., Beaty, D., Kreymer, D., Li, D., Adkins, D., Xu, D., Testuggine, D., David, D., Parikh, D., Liskovich, D., Foss, D., Wang, D., Le, D., Holland, D., Dowling, E., Jamil, E., Montgomery, E., Presani, E., Hahn, E., Wood, E., Le, E.-T., Brinkman, E., Arcaute, E., Dunbar, E., Smothers, E., Sun, F., Kreuk, F., Tian, F., Kokkinos, F., Ozgenel, F., Caggioni, F., Kanayet, F., Seide, F., Florez, G. M., Schwarz, G., Badeer, G., Swee, G., Halpern, G., Herman, G., Sizov, G., Guangyi, Zhang, Lakshminarayanan, G., Inan, H., Shojanazeri, H., Zou, H., Wang, H., Zha, H., Habeeb, H., Rudolph, H., Suk, H., Aspegren, H., Goldman, H., Zhan, H., Damlaj, I., Molybog, I., Tufanov, I., Leontiadis, I., Veliche, I.-E., Gat, I., Weissman, J., Geboski, J., Kohli, J., Lam, J., Asher, J., Gaya, J.-B., Marcus, J., Tang, J., Chan, J., Zhen, J., Reizenstein, J., Teboul, J., Zhong, J., Jin, J., Yang, J., Cummings, J., Carvill, J., Shepard, J., McPhie, J., Torres, J., Ginsburg, J., Wang, J., Wu, K., U, K. H., Saxena, K., Khandelwal, K., Zand, K., Matosich, K., Veeraraghavan, K., Michelena, K., Li, K., Jagadeesh, K., Huang, K., Chawla, K., Huang, K., Chen, L., Garg, L., A, L., Silva, L., Bell, L., Zhang, L., Guo, L., Yu, L., Moshkovich, L., Wehrstedt, L., Khabsa, M., Avalani, M., Bhatt, M., Mankus, M., Hasson, M., Lennie, M., Reso, M., Groshev, M., Naumov, M., Lathi, M., Keneally, M., Liu, M., Seltzer, M. L., Valko, M., Restrepo, M., Patel, M., Vyatskov, M., Samvelyan, M., Clark, M., Macey, M., Wang, M., Hermoso, M. J., Metanat, M., Rastegari, M., Bansal, M., Santhanam, N., Parks, N., White, N., Bawa, N., Singhal, N., Egebo, N., Usunier, N., Mehta, N., Laptev, N. P., Dong, N., Cheng, N., Chernoguz, O., Hart, O., Salpekar, O., Kalinli, O., Kent, P., Parekh, P., Saab, P., Balaji, P., Rittner, P., Bontrager, P., Roux, P., Dollar, P., Zvyagina, P., Ratanchandani, P., Yuvraj, P., Liang, Q., Alao, R., Rodriguez, R., Ayub, R., Murthy, R., Nayani, R., Mitra, R., Parthasarathy, R., Li, R., Hogan, R., Battey, R., Wang, R., Howes, R., Rinott, R., Mehta, S., Siby, S., Bondu, S. J., Datta, S., Chugh, S., Hunt, S., Dhillon, S., Sidorov, S., Pan, S., Mahajan, S., Verma, S., Yamamoto, S., Ramaswamy, S., Lindsay, S., Lindsay, S., Feng, S., Lin, S., Zha, S. C., Patil, S., Shankar, S., Zhang, S., Zhang, S., Wang, S., Agarwal, S., Sajuyigbe, S., Chintala, S., Max, S., Chen, S., Kehoe, S., Satterfield, S., Govindaprasad, S., Gupta, S., Deng, S., Cho, S., Virk, S., Subramanian, S., Choudhury, S., Goldman, S., Remez, T., Glaser, T., Best, T., Koehler, T., Robinson, T., Li, T., Zhang, T., Matthews, T., Chou, T., Shaked, T., Vontimitta, V., Ajayi, V., Montanez, V., Mohan, V., Kumar, V. S., Mangla, V., Ionescu, V., Poenaru, V., Mihailescu, V. T., Ivanov, V., Li, W., Wang, W., Jiang, W., Bouaziz, W., Constable, W., Tang, X., Wu, X., Wang, X., Wu, X., Gao, X., Kleinman, Y., Chen, Y., Hu, Y., Jia, Y., Qi, Y., Li, Y., Zhang, Y., Zhang, Y., Adi, Y., Nam, Y., Yu, Wang, Zhao, Y., Hao, Y., Qian, Y., Li, Y., He, Y., Rait, Z., DeVito, Z., Rosnbrick, Z., Wen, Z., Yang, Z., Zhao, Z., and Ma, Z. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/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. 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), pp. 15789–15809, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.841. URL https://aclanthology.org/2024.acl-long.841/.

Gu, J., Jiang, X., Shi, Z., Tan, H., Zhai, X., Xu, C., Li, W., Shen, Y., Ma, S., Liu, H., Wang, S., Zhang, K., Wang, Y., Gao, W., Ni, L., and Guo, J. A survey on llm-as-a-judge, 2025. URL https://arxiv.org/abs/2411.15594.

Kirchenbauer, J., Geiping, J., Wen, Y., Katz, J., Miers, I., and Goldstein, T. A watermark for large language models. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 17061–17084. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/kirchenbauer23a.html.

Krishna, K., Song, Y., Karpinska, M., Wieting, J. F., and Iyyer, M. Paraphrasing evades detectors of AI-generated text, but retrieval is an effective defense. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=WbFhFvjjKj.

Langlais, P.-C., Chizhov, P., Arnett, C., Hinostroza, C. R., Nee, M., Jones, E. K., Girard, I., Mach, D., Stasenko, A., and Yamshchikov, I. P. Common corpus: The largest collection of ethical data for LLM pre-training. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=0wSlFpMsGb.

Lau, G. K. R., Niu, X., Dao, H., Chen, J., Foo, C.-S., and Low, B. K. H. Waterfall: Scalable framework for robust text watermarking and provenance for LLMs. In Al-Onaizan, Y., Bansal, M., and Chen, Y.-N. (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 20432–20466, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.1138. URL https://aclanthology.org/2024.emnlp-main.1138/.

Lu, X., Wang, J., Zhao, Z., Dai, Z., Foo, C.-S., Ng, S.-K., and Low, B. K. H. WASA: WAtermark-based source attribution for large language model-generated data. In Che, W., Nabende, J., Shutova, E., and Pilehvar, M. T. (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 23791–23824, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.1219. URL https://aclanthology.org/2025.findings-acl.1219/.

Lu, X., Niu, X., Lau, G. K. R., Bui, N., Sim, R. H. L., Himawan, J. R., Wen, F., Foo, C.-S., Ng, S.-K., and Low, B. K. H. Waterdrum: Watermark-based data-centric unlearning metric. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=5GVfneFvhq.

Maini, P., Feng, Z., Schwarzschild, A., Lipton, Z. C., and Kolter, J. Z. TOFU: A task of fictitious unlearning for LLMs. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=B41hNBoWLo.

Miller, G. A. Wordnet: a lexical database for english. *Commun. ACM*, 38(11):39–41, November 1995. ISSN 0001-0782. doi: 10.1145/219717.219748. URL https://doi.org/10.1145/219717.219748.

Mökander, J., Schuett, J., Kirk, H. R., and Floridi, L. Auditing large language models: a three-layered approach. *AI and Ethics*, 4(4):1085–1115, May 2023. ISSN 2730-5961. doi: 10.1007/s43681-023-00289-2. URL http://dx.doi.org/10.1007/s43681-023-00289-2.

OpenAI, :, Hurst, A., Lerer, A., Goucher, A. P., Perelman, A., Ramesh, A., Clark, A., Ostrow, A., Welihinda, A., Hayes, A., Radford, A., Madry, A., Baker-Whitcomb, A., Beutel, A., Borzunov, A., Carney, A., Chow, A., Kirillov, A., Nichol, A., Paino, A., Renzin, A., Passos, A. T., Kirillov, A., Christakis, A., Conneau, A., Kamali, A., Jabri, A., Moyer, A., Tam, A., Crookes, A., Tootoochian, A., Tootoonchian, A., Kumar, A., Vallone, A., Karpathy, A., Braunstein, A., Cann, A., Codispoti, A., Galu, A., Kondrich, A., Tulloch, A., Mishchenko, A., Baek, A., Jiang, A., Pelisse, A., Woodford, A., Gosalia, A., Dhar, A., Pantuliano, A., Nayak, A., Oliver, A., Zoph, B., Ghorbani, B., Leimberger, B., Rossen, B., Sokolowsky, B., Wang, B., Zweig, B., Hoover, B., Samic, B., McGrew, B., Spero, B., Giertler, B., Cheng, B., Lightcap, B., Walkin, B., Quinn, B., Guarraci, B., Hsu, B., Kellogg, B., Eastman, B., Lugaresi, C., Wainwright, C., Bassin, C., Hudson, C., Chu, C., Nelson, C., Li, C., Shern, C. J., Conger, C., Barette, C., Voss, C., Ding, C., Lu, C., Zhang, C., Beaumont, C., Hallacy, C., Koch, C., Gibson, C., Kim, C., Choi, C., McLeavey, C., Hesse, C., Fischer, C., Winter, C., Czarnecki, C., Jarvis, C., Wei, C., Koumouzelis, C., Sherburn, D., Kappler, D., Levin, D., Levy, D., Carr, D., Farhi, D., Mely, D., Robinson, D., Sasaki, D., Jin, D., Valladares, D., Tsipras, D., Li, D., Nguyen, D. P., Findlay, D., Oiwoh, E., Wong, E., Asdar, E., Proehl, E., Yang, E., Antonow, E., Kramer, E., Peterson, E., Sigler, E., Wallace, E., Brevdo, E., Mays, E., Khorasani, F., Such, F. P., Raso, F., Zhang, F., von Lohmann, F., Sulit, F., Goh, G., Oden, G., Salmon,

G., Starace, G., Brockman, G., Salman, H., Bao, H., Hu, H., Wong, H., Wang, H., Schmidt, H., Whitney, H., Jun, H., Kirchner, H., de Oliveira Pinto, H. P., Ren, H., Chang, H., Chung, H. W., Kivlichan, I., O'Connell, I., O'Connell, I., Osband, I., Silber, I., Sohl, I., Okuyucu, I., Lan, I., Kostrikov, I., Sutskever, I., Kanitscheider, I., Gulrajani, I., Coxon, J., Menick, J., Pachocki, J., Aung, J., Betker, J., Crooks, J., Lennon, J., Kiros, J., Leike, J., Park, J., Kwon, J., Phang, J., Teplitz, J., Wei, J., Wolfe, J., Chen, J., Harris, J., Varavva, J., Lee, J. G., Shieh, J., Lin, J., Yu, J., Weng, J., Tang, J., Yu, J., Jang, J., Candela, J. Q., Beutler, J., Landers, J., Parish, J., Heidecke, J., Schulman, J., Lachman, J., McKay, J., Uesato, J., Ward, J., Kim, J. W., Huizinga, J., Sitkin, J., Kraaijeveld, J., Gross, J., Kaplan, J., Snyder, J., Achiam, J., Jiao, J., Lee, J., Zhuang, J., Harriman, J., Fricke, K., Hayashi, K., Singhal, K., Shi, K., Karthik, K., Wood, K., Rimbach, K., Hsu, K., Nguyen, K., Gu-Lemberg, K., Button, K., Liu, K., Howe, K., Muthukumar, K., Luther, K., Ahmad, L., Kai, L., Itow, L., Workman, L., Pathak, L., Chen, L., Jing, L., Guy, L., Fedus, L., Zhou, L., Mamitsuka, L., Weng, L., McCallum, L., Held, L., Ouyang, L., Feuvrier, L., Zhang, L., Kondraciuk, L., Kaiser, L., Hewitt, L., Metz, L., Doshi, L., Aflak, M., Simens, M., Boyd, M., Thompson, M., Dukhan, M., Chen, M., Gray, M., Hudnall, M., Zhang, M., Aljubeh, M., Litwin, M., Zeng, M., Johnson, M., Shetty, M., Gupta, M., Shah, M., Yatbaz, M., Yang, M. J., Zhong, M., Glaese, M., Chen, M., Janner, M., Lampe, M., Petrov, M., Wu, M., Wang, M., Fradin, M., Pokrass, M., Castro, M., de Castro, M. O. T., Pavlov, M., Brundage, M., Wang, M., Khan, M., Murati, M., Bavarian, M., Lin, M., Yesildal, M., Soto, N., Gimelshein, N., Cone, N., Staudacher, N., Summers, N., LaFontaine, N., Chowdhury, N., Ryder, N., Stathas, N., Turley, N., Tezak, N., Felix, N., Kudige, N., Keskar, N., Deutsch, N., Bundick, N., Puckett, N., Nachum, O., Okelola, O., Boiko, O., Murk, O., Jaffe, O., Watkins, O., Godement, O., Campbell-Moore, O., Chao, P., McMillan, P., Belov, P., Su, P., Bak, P., Bakkum, P., Deng, P., Dolan, P., Hoeschele, P., Welinder, P., Tillet, P., Pronin, P., Tillet, P., Dhariwal, P., Yuan, Q., Dias, R., Lim, R., Arora, R., Troll, R., Lin, R., Lopes, R. G., Puri, R., Miyara, R., Leike, R., Gaubert, R., Zamani, R., Wang, R., Donnelly, R., Honsby, R., Smith, R., Sahai, R., Ramchandani, R., Huet, R., Carmichael, R., Zellers, R., Chen, R., Chen, R., Nigmatullin, R., Cheu, R., Jain, S., Altman, S., Schoenholz, S., Toizer, S., Miserendino, S., Agarwal, S., Culver, S., Ethersmith, S., Gray, S., Grove, S., Metzger, S., Hermani, S., Jain, S., Zhao, S., Wu, S., Jomoto, S., Wu, S., Shuaiqi, Xia, Phene, S., Papay, S., Narayanan, S., Coffey, S., Lee, S., Hall, S., Balaji, S., Broda, T., Stramer, T., Xu, T., Gogineni, T., Christianson, T., Sanders, T., Patwardhan, T., Cunninghman, T., Degry, T., Dimson, T., Raoux, T., Shadwell, T., Zheng, T., Underwood, T., Markov, T., Sherbakov, T., Rubin, T., Stasi, T., Kaftan, T., Heywood, T., Peterson, T., Walters, T., Eloundou, T., Qi, V., Moeller, V., Monaco, V., Kuo, V., Fomenko, V., Chang, W., Zheng, W., Zhou, W., Manassra, W., Sheu, W., Zaremba, W., Patil, Y., Qian, Y., Kim, Y., Cheng, Y., Zhang, Y., He, Y., Zhang, Y., Jin, Y., Dai, Y., and Malkov, Y. Gpt-4o system card, 2024. URL https://arxiv.org/abs/2410.21276.

Pan, L., Liu, A., He, Z., Gao, Z., Zhao, X., Lu, Y., Zhou, B., Liu, S., Hu, X., Wen, L., King, I., and Yu, P. S. MarkLLM: An open-source toolkit for LLM watermarking. In Hernandez Farias, D. I., Hope, T., and Li, M. (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 61–71, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-demo.7. URL https://aclanthology.org/2024.emnlp-demo.7/.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1), January 2020. ISSN 1532-4435.

Rastogi, S., Maini, P., and Pruthi, D. STAMP your content: Proving dataset membership via watermarked rephrasings. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=qF6mxani2X.

Sander, T., Fernandez, P., Durmus, A. O., Douze, M., and Furon, T. Watermarking makes language models radioactive. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=qGiZQb1Khm.

Schler, J., Koppel, M., Argamon, S., and Pennebaker, J. Effects of age and gender on blogging. In *Proceedings of 2006 AAAI Spring Symposium on Computational Approaches for Analyzing Weblogs*, pp. 199–205, 01 2006.

Shetty, P., Haque, M., Babkin, P., Ma, Z., Liu, X., and Veloso, M. Perturb your data: Paraphrase-guided training data watermarking. In Koenig, S., Jenkins, C., and Taylor, M. E. (eds.), *Fortieth AAAI Conference on Artificial Intelligence, Thirty-Eighth Conference on Innovative Applications of Artificial Intelligence, Sixteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2026, Singapore, January 20-27, 2026*, pp. 32938–32946. AAAI Press, 2026. doi: 10.1609/AAAI.V40I39.40575. URL https://doi.org/10.1609/aaai.v40i39.40575.

Shi, W., Ajith, A., Xia, M., Huang, Y., Liu, D., Blevins, T., Chen, D., and Zettlemoyer, L. Detecting pretraining data from large language models. In *The Twelfth*

International Conference on Learning Representations, 2024. URL https://openreview.net/forum?id=zWqr3MQuNs.

Shi, W., Lee, J., Huang, Y., Malladi, S., Zhao, J., Holtzman, A., Liu, D., Zettlemoyer, L., Smith, N. A., and Zhang, C. MUSE: Machine unlearning six-way evaluation for language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=TArmA033BU.

Shokri, R., Stronati, M., Song, C., and Shmatikov, V. Membership Inference Attacks Against Machine Learning Models . In *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 3–18, Los Alamitos, CA, USA, May 2017. IEEE Computer Society. doi: 10.1109/SP.2017.41. URL https://doi.ieeecomputersociety.org/10.1109/SP.2017.41.

Song, K., Tan, X., Qin, T., Lu, J., and Liu, T.-Y. Mpnet: Masked and permuted pre-training for language understanding. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 16857–16867. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/c3a690be93aa602ee2dc0ccab5b7b67e-Paper.pdf.

Team, G., Riviere, M., Pathak, S., Sessa, P. G., Hardin, C., Bhupatiraju, S., Hussenot, L., Mesnard, T., Shahriari, B., Ramé, A., Ferret, J., Liu, P., Tafti, P., Friesen, A., Casbon, M., Ramos, S., Kumar, R., Lan, C. L., Jerome, S., Tsitsulin, A., Vieillard, N., Stanczyk, P., Girgin, S., Momchev, N., Hoffman, M., Thakoor, S., Grill, J.-B., Neyshabur, B., Bachem, O., Walton, A., Severyn, A., Parrish, A., Ahmad, A., Hutchison, A., Abdagic, A., Carl, A., Shen, A., Brock, A., Coenen, A., Laforge, A., Paterson, A., Bastian, B., Piot, B., Wu, B., Royal, B., Chen, C., Kumar, C., Perry, C., Welty, C., Choquette-Choo, C. A., Sinopalnikov, D., Weinberger, D., Vijaykumar, D., Rogozińska, D., Herbison, D., Bandy, E., Wang, E., Noland, E., Moreira, E., Senter, E., Eltyshev, E., Visin, F., Rasskin, G., Wei, G., Cameron, G., Martins, G., Hashemi, H., Klimczak-Plucińska, H., Batra, H., Dhand, H., Nardini, I., Mein, J., Zhou, J., Svensson, J., Stanway, J., Chan, J., Zhou, J. P., Carrasqueira, J., Iljazi, J., Becker, J., Fernandez, J., van Amersfoort, J., Gordon, J., Lipschultz, J., Newlan, J., yeong Ji, J., Mohamed, K., Badola, K., Black, K., Millican, K., McDonell, K., Nguyen, K., Sodhia, K., Greene, K., Sjoesund, L. L., Usui, L., Sifre, L., Heuermann, L., Lago, L., McNealus, L., Soares, L. B., Kilpatrick, L., Dixon, L., Martins, L., Reid, M., Singh, M., Iverson, M., Görner, M., Velloso, M., Wirth, M., Davidow, M., Miller, M., Rahtz, M., Watson, M., Risdal, M., Kazemi, M., Moynihan, M., Zhang, M., Kahng, M., Park, M., Rahman, M., Khatwani, M., Dao, N., Bardoliwalla, N., Devanathan, N., Dumai, N., Chauhan, N., Wahltinez, O., Botarda, P., Barnes, P., Barham, P., Michel, P., Jin, P., Georgiev, P., Culliton, P., Kuppala, P., Comanescu, R., Merhej, R., Jana, R., Rokni, R. A., Agarwal, R., Mullins, R., Saadat, S., Carthy, S. M., Cogan, S., Perrin, S., Arnold, S. M. R., Krause, S., Dai, S., Garg, S., Sheth, S., Ronstrom, S., Chan, S., Jordan, T., Yu, T., Eccles, T., Hennigan, T., Kocisky, T., Doshi, T., Jain, V., Yadav, V., Meshram, V., Dharmadhikari, V., Barkley, W., Wei, W., Ye, W., Han, W., Kwon, W., Xu, X., Shen, Z., Gong, Z., Wei, Z., Cotruta, V., Kirk, P., Rao, A., Giang, M., Peran, L., Warkentin, T., Collins, E., Barral, J., Ghahramani, Z., Hadsell, R., Sculley, D., Banks, J., Dragan, A., Petrov, S., Vinyals, O., Dean, J., Hassabis, D., Kavukcuoglu, K., Farabet, C., Buchatskaya, E., Borgeaud, S., Fiedel, N., Joulin, A., Kenealy, K., Dadashi, R., and Andreev, A. Gemma 2: Improving open language models at a practical size, 2024. URL https://arxiv.org/abs/2408.00118.

Tu, S., Sun, Y., Bai, Y., Yu, J., Hou, L., and Li, J. WaterBench: Towards holistic evaluation of watermarks for large 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)*, pp. 1517–1542, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.83. URL https://aclanthology.org/2024.acl-long.83/.

Xu, X., Wu, Z., Qiao, R., Verma, A., Shu, Y., Wang, J., Niu, X., He, Z., Chen, J., Zhou, Z., Lau, G. K. R., Dao, H., Agussurja, L., Sim, R. H. L., Lin, X., Hu, W., Dai, Z., Koh, P. W., and Low, B. K. H. Position paper: Data-centric AI in the age of large language models. In Al-Onaizan, Y., Bansal, M., and Chen, Y.-N. (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 11895–11913, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.695. URL https://aclanthology.org/2024.findings-emnlp.695/.

Yao, Y., Xu, X., and YangLiu. Large language model unlearning. In Globerson, A., Mackey, L., Belgrave, D., Fan, A., Paquet, U., Tomczak, J., and Zhang, C. (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 105425–105475. Curran Associates, Inc., 2024. doi: 10.52202/079017-3346. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/be52acf6bccf4a8c0a90fe2f5cfcead3-Paper-Conference.pdf.

Zhang, J., Zhao, Y., Saleh, M., and Liu, P. PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization. In III, H. D. and Singh, A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 11328–11339. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/zhang20ae.html.

Zhang, J., Sun, J., Yeats, E., Ouyang, Y., Kuo, M., Zhang, J., Yang, H. F., and Li, H. Min-k%++: Improved baseline for pre-training data detection from large language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=ZGkfoufDaU.

Zhang, R., Lin, L., Bai, Y., and Mei, S. Negative preference optimization: From catastrophic collapse to effective unlearning. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=MXLBXjQkmb.

Zhao, X., Ananth, P. V., Li, L., and Wang, Y.-X. Provable robust watermarking for AI-generated text. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=SsmT8aO45L.

A. Datasets

In this section, we provide more details on the curation of our datasets used in this project and the reason for the design choices.

A.1. C4 realnewslike

The original C4 realnewslike dataset has roughly 13 million samples of news articles. Since we are interested in data-centric watermarking, we parse the `url` column of the C4 realnewslike dataset (Raffel et al., 2020) to extract the institution that produced a given news article. Then, to ensure the quality of the news article, we first rank the top 50 institutions by article volume as given in Table 6. And then, we randomly sample 10 institutions to be used for our dataset, which resulted in `https://aljazeera.com/`, `https://cnbc.com/`, `https://nytimes.com/`, `https://reuters.com/`, `https://theatlantic.com/`, `https://npr.org/`, `https://latimes.com/`, `https://bbc.com/`, `https://theguardian.com/`, and `https://wired.com/`. 1,000 news articles are extracted per institution to build our dataset of size 10,000.

Afterwards, we filter the text to only include $300 \leq words \leq 2500$ to ensure that the text is sufficiently long enough to realistically absorb a watermark while not being too long such that it exceeds the context length of open-source LLMs.

Rank	Institution	Count	Rank	Institution	Count
1	reuters.com	184,891	26	usatoday.com	57,751
2	nytimes.com	172,510	27	nypost.com	57,637
3	latimes.com	143,984	28	npr.org	57,337
4	theguardian.com	135,552	29	time.com	57,003
5	businessinsider.com	104,965	30	baltimoresun.com	56,164
6	zdnet.com	92,566	31	stanford.edu	53,903
7	bbc.co.uk	91,266	32	mashable.com	53,886
8	forbes.com	90,558	33	gizmodo.com	53,474
9	foxnews.com	87,269	34	hindustantimes.com	52,547
10	cnet.com	85,451	35	cnbc.com	52,287
11	chicagotribune.com	83,732	36	azcentral.com	50,441
12	washingtonpost.com	82,105	37	adweek.com	47,683
13	aljazeera.com	79,616	38	csmonitor.com	47,671
14	telegraph.co.uk	79,528	39	thesun.co.uk	47,343
15	cnn.com	78,388	40	edweek.org	46,671
16	dailymail.co.uk	78,275	41	nydailynews.com	45,898
17	indiatimes.com	72,990	42	chron.com	45,762
18	express.co.uk	72,761	43	straitstimes.com	45,167
19	theatlantic.com	69,857	44	dailystar.co.uk	44,804
20	sfgate.com	65,764	45	techcrunch.com	43,577
21	foxbusiness.com	63,544	46	pcmag.com	43,146
22	fool.com	63,499	47	inquisitr.com	43,094
23	rt.com	63,085	48	cbsnews.com	42,850
24	ndtv.com	60,731	49	deadline.com	42,742
25	bbc.com	59,581	50	wired.com	42,379

Table 6. Top 50 Institutions by Article Volume in C4 RealNewsLike.

A.2. Blog Authorship Corpus

The original Blog Authorship Corpus (Schler et al., 2006) contains posts from 19,320 bloggers with a total of 681,288 blog posts. This dataset already has a column `id` as a unique identifier for each blogger which nicely suits our needs. Since realistically there are more bloggers than newspapers on the web, we select randomly select 20 unique id's from the dataset. A blogger would also produce fewer content compared to entire newspapers, so we choose 500 blog posts for each blogger, towards a dataset of size 10,000. Blog posts tend to be much shorter compared to news articles, so we also took this into account during the filtering process. We select blog posts with $100 \leq words \leq 1500$ to make sure the text is long enough to absorb the watermark and still fits within the LLM's context length.

B. Model-centric Watermarking

We summarise the five model-centric schemes we adapt to the data-centric setting; the same algorithm is applied at watermarking time and the same detector at audit time.

KGW (Kirchenbauer et al., 2023): per-step partition of the vocabulary into a key-dependent "green" subset (γ=0.25) and

complementary "red" subset, with logit bias δ on green. Detector recomputes the partition and reports a z-score on the green-token fraction.

Unigram (Zhao et al., 2024): KGW with a single corpus-level partition rather than per-position. Cheaper detector, lower per-token paraphrase robustness.

Waterfall (Lau et al., 2024): paraphrase-and-rerank — an LLM rewrites the source under group beam search, and the candidate maximising key-conditioned similarity-plus-watermark-score is emitted.

SynthID (Dathathri et al., 2024): tournament-style PRF over the next-token distribution. Non-distortionary in expectation, which makes the signal hard to recover after fine-tuning.

EXP-Gumbel (Aaronson, 2022): key-conditioned Gumbel perturbation on the logits. Like SynthID, the signal is in the random draws; unlike SynthID, per-row signal density is close to KGW's.

All five share a common `BaseWatermark` interface in our released code.

C. Data Provenance Applications

We expand here on the three audit tasks of Section 2.2.

C.1. Data Membership Detection

Binary membership: given key $k^{(i)}$ and prompts $\mathcal{Q}_i$, score continuations under the key-specific detector, aggregate $A_i = \mathrm{Agg}_q\, s_i(q, M_{\mathcal{T}}(q))$, threshold at τ_i. The classical analogue is MIA on raw text; under verbatim publication MIA achieves higher AUROC (≈ 1.0) at the cost of forcing the data owner to expose unmodified content. Under the publish-paraphrased adversary the MIA route collapses to ≈ 0.5 across all four detectors (`loss`, `min_k`, `min_k_pp`, `zlib_ratio`) while watermarking holds at 0.65–0.84 AUROC — the regime where data-centric watermarking is the only viable audit primitive.

MIA detector choice. Table 7 evaluates the four MIA detectors on the un-fine-tuned base models against the same seen / unseen sets used for the FT-model audits. Anything ≈ 0.5 on the base $\Rightarrow$ the detector is membership-driven; anything $> 0.5 \Rightarrow$ an intrinsic text-difficulty confound that inflates the FT-model AUROC by roughly that gap. `min_k_pp` is the only detector that hits chance on both bases (0.501, 0.485) — the bias-corrected formulation (Zhang et al., 2025) divides out per-position vocabulary entropy, which is exactly what makes it well-calibrated. `loss` and `min_k` carry ~ 0.07–0.08 residual confound; `zlib_ratio` is broken in this protocol (base AUROC 0.88). We therefore use `min_k_pp` as the headline MIA baseline throughout the paper.

Cell	loss	min_k	min_k_pp	zlib_ratio
Llama-3.1-8B-Instruct (base)	0.571	0.582	**0.501**	0.883
Gemma-2-9b-it (base)	0.560	0.560	**0.485**	0.881

Table 7. Reference-model MIA AUC on un-fine-tuned base models, C4 realnewslike.

Tables 20 and 9 report Setting B AUROC on the same audit, but applied to vanilla-FT models (the data owner publishes raw `original_text` and the model trains directly on it — no watermark). `min_k_pp` saturates at 1.000 on Blog across all four base models, confirming that verbatim memorization in a 10k $\times$ 3-epoch fine-tune is effectively perfect when the model trains directly on the raw text. C4 sits slightly off-ceiling at 0.97–0.99 across detectors. The contrast against §F.2 / §F.3 (model-side AUROC under watermarking) is the qualitative privacy trade-off: Setting B beats best-Setting-A by 0.01–0.14, but only at the cost of forcing the data owner to publish raw content.

C.2. Source Attribution

K-class identification: score the same continuations under each candidate's detector s_j, aggregate A_j, predict $\hat{\jmath} = \arg\max_j A_j$. We report per-id pooled accuracy — scores averaged across all prompts of true owner j before the argmax — as the realistic-audit metric: a data owner running the audit has all evidence rows in hand, and pooling sharpens the argmax even when per-row signal is noisy.

Model	loss	min_k	min_k_pp	zlib_ratio
Llama-8B	0.9800	0.9818	0.9845	0.9783
Qwen-7B	0.9774	0.9794	0.9835	0.9542
Qwen-14B	0.9769	0.9782	0.9803	0.9569
Gemma-9B	0.9830	0.9832	0.9848	0.9835

Table 8. Setting B (vanilla MIA) AUROC on the C4 realnewslike dataset. Each row is a base model fine-tuned for 3 epochs on raw `original_text` (no watermark).

Model	loss	min_k	min_k_pp	zlib_ratio
Llama-8B	**1.0000**	**1.0000**	**1.0000**	0.9822
Qwen-7B	**1.0000**	**1.0000**	**1.0000**	0.9528
Qwen-14B	**1.0000**	**1.0000**	**1.0000**	0.9619
Gemma-9B	**1.0000**	**1.0000**	**1.0000**	**0.9993**

Table 9. Setting B (vanilla MIA) AUROC on the Blog Authorship Corpus dataset. Each row is a base model fine-tuned for 3 epochs on raw `original_text` (no watermark).

Let $M \in \mathbb{R}^{T\times N}$ be the score matrix obtained by querying the suspect model with T prompts and scoring each continuation under every candidate key $\{k^{(1)}, \ldots, k^{(N)}\}$. With $R_i = \{r : y_r = i\}$ denoting the rows of the evaluation set authored by data owner i, we pool by source and predict

$$\hat{y}_i = \arg\max_{j\in\{1,\ldots,N\}} \tfrac{1}{|R_i|}\sum_{r\in R_i} M_{r,j}, \qquad \text{Attr-Acc} = \tfrac{1}{N}\sum_{i=1}^{N} \mathbb{1}[\hat{y}_i = i]\,. \tag{1}$$

The royalty share assigned to data owner i is then $\text{share}(i) = \sum_r M_{r,i} \big/ \sum_{j,r} M_{r,j}$.

The data-side specificity precondition (Table 11) is necessary but not sufficient: the model must additionally preserve per-key signal in its continuations. Both conditions hold for the distortion-based watermarks (KGW, Unigram, Waterfall) on both base models; attribution clears 0.95 on every C4 cell except Unigram-Llama (0.80).

C.3. Machine Unlearning Evaluation

Given an unlearning procedure U targeting data owner i, we report (i) the reduction $\Delta_i = A_i(M_{\mathcal{T}}) - A_i(M_{-\mathcal{F}})$ and (ii) the residual relative to a reference $R_i = A_i(M_{-\mathcal{F}}) - A_i(M_{\text{ref}})$. The reference is essential: without it, a small $A_i(M_{-\mathcal{F}})$ is ambiguous between successful unlearning and general fluency damage. We use a *retrain-on-retain oracle* as M_{ref} — a fresh fine-tune from the base model on the retain split alone — giving the lowest physically achievable target-owner signal.

Unlearning operators. **Retain-only retrain (oracle).** Fresh fine-tune from pretrained weights on retain keys $\{6, \ldots, 20\}$ (7,500 rows), 3 epochs at the same hyperparameters as the full-FT (FSDP zero-3, lr `2e-5`, cosine, 3% warmup). Used as M_{ref}.

Gradient Difference (GD-T). Starting from $M_{\mathcal{T}}$, minimise

$$\mathcal{L}_{\text{GD}}(\theta) = -\mathcal{L}_{\text{LM}}(\theta;\, \mathcal{D}^{\text{forget}}) + \mathcal{L}_{\text{LM}}(\theta;\, \mathcal{D}^{\text{retain}})$$

for T steps: ascent on forget, descent on retain, equal sampling. We use **GD-200** (effective batch 128 across 4 GPUs); $T = 200$ is the smallest budget at which forget-side A_i approaches the oracle for KGW-Llama, with longer schedules over-damaging retain.

Negative Preference Optimization (NPO). (Zhang et al., 2024)

$$\mathcal{L}_{\text{NPO}}(\theta) = -\tfrac{2}{\beta}\log\sigma\Big(-\beta\log\tfrac{p_\theta(y|x)}{p_{\mathcal{T}}(y|x)}\Big)\,,$$

$(x, y) \in \mathcal{D}^{\text{forget}}$, with $p_{\mathcal{T}}$ frozen. We use β=0.1 for 500 steps. Retain-side preservation comes only from the KL anchor.

All three share forget / retain splits (keys $\{1..5\}$ / $\{6..20\}$), evaluation protocol (50-token prefix, $K{=}20$ audit, 300 prompts per side), and a single seed. Implementation: `data_mark/trainer/unlearn.py` in our released code.

D. Experimental Settings

Cluster: NVIDIA GH200 GPUs, 1,000+ node-hours total. Watermarking and evaluation use single-GPU array jobs; full-FT uses 4-GPU FSDP zero-3 for 3 epochs at lr `2e-5` (cosine, 3% warmup). Tables are single-seed unless noted; seeded re-runs show ± 0.02–0.03 AUROC drift per cell.

D.1. Fidelity

To evaluate the fidelity, we use the following metrics:

STS. We compare the semantic similarity score between the original text and the watermarked text using an independent embedding model `sentence-transformers/all-mpnet-base-v2` (Song et al., 2020).

LLM Judge. We use the LLM-as-judge framework (Gu et al., 2025), using `gpt-4o-mini` (OpenAI et al., 2024) as our judge model. The judge compares the watermarked text against the original text and will rate it between $\{faithful, partial, unfaithful\}$. We aggregate the score as well assigning them with 1, 0.5, and 0 respectively.

D.2. Attacks

Five content-preserving attacks applied before detection: token-level surface edits (Token Drop, Synonym Substitution) and paraphrase rewrites (LLM, DIPPER, PEGASUS).

D.2.1. Token Drop

Random deletion of 10% of tokens. Most data-centric watermarks are position-dependent (KGW reads consecutive-token pairs; Unigram a per-token partition; SynthID/EXP-Gumbel the per-step PRF), so token drop is a natural lower-bound probe. AUROC degrades 0.00–0.05 for distortion-based watermarks and 0.05–0.20 for non-distortionary schemes.

D.2.2. Synonym Substitution

Replace 10% of content tokens with WordNet synonyms (Miller, 1995). Comparable disruption to token drop on natural-language text, slightly stronger on code where most identifiers lack meaningful synonyms.

D.2.3. LLM Paraphraser

Instruction-tuned LLMs rewrite the watermarked text without knowing the watermark scheme, prompted with:

```
You are a careful paraphraser. Rewrite the user's text using different wording
while preserving its meaning, length, and tone. Output only the paraphrase,
with no preamble or explanation.
```

We use Llama-3.1-8B-Instruct as the headline paraphraser. Qwen and Gemma variants behave similarly (mean per-watermark AUROC differs by < 0.05); see Table 10 for the fidelity profile.

D.2.4. Dipper

DIPPER (Krishna et al., 2023) maximises rewrite diversity while preserving semantics. Default settings; this is the most aggressive paraphrase in our suite (STS ≈ 0.72 vs LLM paraphrasers' ≈ 0.93) and the worst-case adversarial benchmark.

D.2.5. Pegasus

PEGASUS (Zhang et al., 2020) is an abstractive-summarisation model used. Most surface-disruptive (STS ≈ 0.57) but inflates perplexity $3.1\times$ on our independent oracle — a real adversary would not deploy this. Reported for comparability only.

D.2.6. PARAPHRASER FIDELITY COMPARISON

Table 10 reports surface-form preservation for each attack, averaged over all 40 watermarked corpora. STS uses `all-mpnet-base-v2` (Song et al., 2020) between paraphrased text and (i) the watermarked source, (ii) the un-watermarked `original_text`. PPL ratio uses `OLMo-2-1124-7B` (Groeneveld et al., 2024) as an independent oracle. Per-watermark variance is < 0.01 in STS and < 0.05 in PPL ratio, so the profile is a property of the paraphraser.

Paraphraser	STS(wm, attacked)	STS(orig, attacked)	PPL(attacked) / PPL(wm)
LLM paraphrasers			
llama_paraphrase	**0.930**	**0.838**	**0.784**
qwen_paraphrase	**0.933**	**0.841**	1.199
gemma_paraphrase	**0.929**	**0.839**	**0.964**
Dedicated paraphrasers			
pegasus	0.568	0.540	3.122
dipper	0.716	0.670	1.526

Table 10. Paraphraser fidelity comparison, averaged over the 40-corpus benchmark. STS via `sentence-transformers/all-mpnet-base-v2`; PPL ratio via `allenai/OLMo-2-1124-7B`.

Why we report Llama-paraphrase. The three LLM paraphrasers preserve essentially the same semantic content (STS $\in [0.929, 0.933]$); the differentiator is fluency. Llama's PPL ratio is 0.784 (rewrites are slightly *more* fluent than the source), vs. Qwen 1.199 and Gemma 0.964. Llama is therefore the strongest realistic adversary: most fluent rewrite tracking source semantics, which a defender cannot reject on quality grounds. We use it as the canonical "Paraphrase" column and treat Qwen / Gemma as sensitivity tests.

The dedicated paraphrasers and the LLM paraphrasers bound the realistic–worst-case interval. PEGASUS aggressively rewrites surface form (STS 0.568) but inflates perplexity $3.1\times$, easy to reject on fluency screening. DIPPER simultaneously degrades surface form (STS 0.716) and preserves usable fluency (PPL ratio 1.526) — the worst-case adversary in our suite.

E. Other Experimental Results on C4

E.1. Per-id Specificity

Watermark	Model	AUROC
Waterfall	Llama-8B	**1.000**
	Gemma-9B	**1.000**
KGW	Llama-8B	0.998
	Gemma-9B	0.989
Unigram	Llama-8B	0.998
	Gemma-9B	0.996
SynthID	Llama-8B	0.995
	Gemma-9B	0.982
EXP-Gumbel	Llama-8B	**1.000**
	Gemma-9B	0.996

Table 11. AUROC for the per-id separability on the C4 realnewslike dataset.

E.2. Attribution

Watermark	Model	Attribution Accuracy
Waterfall	Llama-8B	**1.000**
	Gemma-9B	**1.000**
KGW	Llama-8B	**1.000**
	Gemma-9B	**1.000**
Unigram	Llama-8B	0.800
	Gemma-9B	**1.000**
SynthID	Llama-8B	**1.000**
	Gemma-9B	**1.000**
EXP-Gumbel	Llama-8B	**1.000**
	Gemma-9B	**1.000**

Table 12. Attribution accuracy on the C4 realnewslike dataset.

E.3. Quality

Table 13 reports STS and LLM-judge fidelity for the C4 realnewslike dataset. STS clusters at 0.88–0.92 across all five watermarks; the LLM judge labels 19–31% of samples as faithful and the remainder as partial, with unfaithful rates below 3% on every cell. Soft fidelity sits at 0.59–0.66, lower than Blog (0.71–0.79) because C4 articles are longer and contain more domain-specific terminology that an LLM rewrite is more likely to drift on.

			LLM Judge			
Watermark	**Model**	**STS**	**Faithful (%)**	**Partial (%)**	**Unfaithful (%)**	**Soft Fidelity**
Waterfall	Llama-8B	**0.917**	19.7	77.7	2.7	0.585
	Gemma-9B	0.897	21.3	77.7	1.0	0.602
KGW	Llama-8B	0.904	**31.3**	67.7	1.0	0.652
	Gemma-9B	0.881	25.7	73.7	0.7	0.625
Unigram	Llama-8B	0.901	26.7	71.3	2.0	0.623
	Gemma-9B	0.879	26.0	72.3	1.7	0.622
SynthID	Llama-8B	0.899	**31.3**	68.7	0.0	**0.657**
	Gemma-9B	0.877	24.7	75.0	0.3	0.622
EXP-Gumbel	Llama-8B	0.900	30.3	69.7	0.0	0.652
	Gemma-9B	0.878	22.0	**78.0**	0.0	0.610

Table 13. Fidelity metrics (STS, LLM Judge) for the C4 realnewslike dataset. Lower Unfaithful is better; higher STS / Faithful / Soft Fidelity is better.

E.4. Detection at Fixed FPR (Data-Side)

Tables 14–16 report data-side TPR@$\{1, 5, 10\}$%FPR on C4 before and after attacks. KGW and Waterfall reach 1.0000 on clean text and survive token-drop / synonym-substitution at ≥ 0.99 across all base models. Llama-paraphrase is the strongest attack, dropping TPR@1% to 0.21–0.99 across cells; Waterfall-Llama at 0.99 is the only configuration that retains near-ceiling low-FPR performance under realistic paraphrase.

E.5. Detection at Fixed FPR (Model-Side)

Tables 17–19 report TPR@$\{1, 5, 10\}$%FPR for the model-side audit on 300 held-out prefixes ($K = 10$, prefix length 100, single seed), complementing the AUROC in Table 2.

Watermark	Model	Before Attack	After Attack		
			Tok. Drop	Syn. Sub.	Paraphrase
Waterfall	Llama-8B	**1.0000**	**1.0000**	**1.0000**	**0.9900**
	Gemma-9B	**1.0000**	**1.0000**	**1.0000**	0.7500
KGW	Llama-8B	**1.0000**	**1.0000**	**1.0000**	0.8567
	Gemma-9B	**1.0000**	0.9620	0.9560	0.3967
Unigram	Llama-8B	0.8160	0.7420	0.7320	0.4733
	Gemma-9B	0.6080	0.4300	0.4360	0.2100
SynthID	Llama-8B	0.9920	0.8800	0.9100	0.5667
	Gemma-9B	0.9600	0.7320	0.8040	0.3033
EXP-Gumbel	Llama-8B	0.9920	0.9580	0.9740	0.6167
	Gemma-9B	0.9360	0.6640	0.7280	0.1867

Table 14. Data-side detector TPR@1%FPR on the C4 realnewslike dataset before and after attacks. Paraphrase reports TPR under the Llama-3.1-8B-Instruct paraphraser.

Watermark	Model	Before Attack	After Attack		
			Tok. Drop	Syn. Sub.	Paraphrase
Waterfall	Llama-8B	**1.0000**	**1.0000**	**1.0000**	**0.9933**
	Gemma-9B	**1.0000**	**1.0000**	**1.0000**	0.8600
KGW	Llama-8B	**1.0000**	**1.0000**	**1.0000**	0.9567
	Gemma-9B	**1.0000**	**1.0000**	0.9980	0.6867
Unigram	Llama-8B	0.9180	0.8540	0.8340	0.6067
	Gemma-9B	0.8520	0.7120	0.6920	0.3867
SynthID	Llama-8B	0.9960	0.9420	0.9620	0.6900
	Gemma-9B	0.9880	0.8240	0.8960	0.3900
EXP-Gumbel	Llama-8B	0.9980	0.9860	0.9900	0.7633
	Gemma-9B	0.9740	0.8660	0.8960	0.3500

Table 15. Data-side detector TPR@5%FPR on the C4 realnewslike dataset before and after attacks. Paraphrase reports TPR under the Llama-3.1-8B-Instruct paraphraser.

E.6. Comparison Against MIA Baselines

We also evaluate the effectiveness of MIA to identify whether a given text is included in the training data. We fine-tune the base LLM on the unwatermarked corpus. The MIA distinguishes between members (e.g. training data) vs non-members (e.g. holdout set).

Model	loss	min_k	min_k_pp	zlib_ratio
Llama-8B	0.9800	0.9818	0.9845	0.9783
Gemma-9B	0.9830	0.9832	**0.9848**	0.9835

Table 20. AUROC on the C4 realnewslike dataset using MIA. Each row is a base model fine-tuned for 3 epochs on raw text.

E.7. Impact on Model Utility

F. Experimental Results on Other Datasets

The main paper focuses on C4 realnewslike since many data-provenance lawsuits involve newspapers. We additionally report Blog and code-domain (MBPP/MBJSP) results to characterise generalisation across content types.

Watermark	Model	Before Attack	After Attack		
			Tok. Drop	Syn. Sub.	Paraphrase
Waterfall	Llama-8B	**1.0000**	**1.0000**	**1.0000**	**1.0000**
	Gemma-9B	**1.0000**	**1.0000**	**1.0000**	0.9033
KGW	Llama-8B	**1.0000**	**1.0000**	**1.0000**	0.9800
	Gemma-9B	**1.0000**	**1.0000**	**1.0000**	0.7800
Unigram	Llama-8B	0.9500	0.9040	0.8840	0.6933
	Gemma-9B	0.9420	0.8340	0.8340	0.5400
SynthID	Llama-8B	0.9960	0.9680	0.9760	0.7333
	Gemma-9B	0.9900	0.8760	0.9280	0.4467
EXP-Gumbel	Llama-8B	0.9980	0.9940	0.9920	0.8367
	Gemma-9B	0.9860	0.9340	0.9380	0.4400

Table 16. Data-side detector TPR@10%FPR on the C4 realnewslike dataset before and after attacks. Paraphrase reports TPR under the Llama-3.1-8B-Instruct paraphraser.

Watermark	Model	TPR@1%FPR
Waterfall	Llama-8B	0.440
	Gemma-9B	0.833
KGW	Llama-8B	0.367
	Gemma-9B	0.733
Unigram	Llama-8B	0.343
	Gemma-9B	0.663
SynthID	Llama-8B	0.060
	Gemma-9B	0.403
EXP-Gumbel	Llama-8B	0.080
	Gemma-9B	0.533

Table 17. Model-side detection TPR@1%FPR on the C4 realnewslike dataset with $K{=}10$ candidate keys.

F.1. Blog Authorship Corpus

Blog Authorship Corpus (Schler et al., 2006) uses 20 data owners with 500 posts each. Model-side audit uses $K = 20$ candidate keys.

F.1.1. Effectiveness of Watermark on Corpus

Detection and Robustness to Attacks.

All five watermarks achieve near-ceiling clean AUROC (≥ 0.96 on both bases). Cheap perturbations cost 0.01–0.08. Llama-paraphrase is the strongest attack, dropping AUROC to 0.57–0.96; Waterfall-Llama retains the most signal (0.9610). Gemma corpora are uniformly more paraphrase-vulnerable than Llama corpora, mirroring C4.

Does Watermarking Degrade Text Quality?

STS clusters at 0.83–0.86. LLM judge: 43–58% faithful, $< 2\%$ unfaithful on every cell, soft fidelity 0.71–0.79. No watermark dominates on quality, so the choice between watermarks on the Blog Authorship Corpus is a robustness trade-off, not a quality trade-off.

Distinguishing Between Multiple Data Owners.

All five watermarks reach mean per-key AUROC ≥ 0.97 on both bases (Waterfall: exactly 1.000); the data-side specificity precondition for the $K = 20$ model-side audit holds.

Watermark	Model	TPR@5%FPR
Waterfall	Llama-8B	0.653
	Gemma-9B	0.883
KGW	Llama-8B	0.560
	Gemma-9B	0.883
Unigram	Llama-8B	0.457
	Gemma-9B	0.807
SynthID	Llama-8B	0.153
	Gemma-9B	0.540
EXP-Gumbel	Llama-8B	0.230
	Gemma-9B	0.727

Table 18. Model-side detection TPR@5%FPR on the C4 realnewslike dataset with $K=10$ candidate keys.

Watermark	Model	TPR@10%FPR
Waterfall	Llama-8B	0.767
	Gemma-9B	0.923
KGW	Llama-8B	0.697
	Gemma-9B	0.913
Unigram	Llama-8B	0.520
	Gemma-9B	0.863
SynthID	Llama-8B	0.260
	Gemma-9B	0.607
EXP-Gumbel	Llama-8B	0.350
	Gemma-9B	0.767

Table 19. Model-side detection TPR@10%FPR on the C4 realnewslike dataset with $K=10$ candidate keys.

F.1.2. Watermark Persistence through Training

Are the Watermarks Really Radioactive?

Tables 26, 27, and 28 report the corresponding model-side TPR at fixed FPR operating points for the Blog corpus.

Persistence on Blog is measurably weaker than C4: per-cell AUROC 0.64–0.94 vs. 0.61–0.99. Drivers: blog posts are shorter (median $\sim$250 words vs. $\sim$500+), and 20-way attribution is statistically harder than 10-way. Every cell still exceeds chance, and Waterfall/KGW cluster at ≥ 0.89 on both bases.

F.1.3. Detection at Fixed FPR

Tables 29–31 report data-side TPR@$\{1, 5, 10\}$%FPR — the realistic deployment regime under a fixed false-accusation budget.

TPR@1% sharpens the cross-watermark contrast: KGW and Waterfall stay at 1.000 clean and ≥ 0.99 under cheap perturbations on Llama-8B, but paraphrase TPR@1% drops to 0.05–0.71. Gemma cells lose systematically more than Llama; Unigram-Gemma at 0.05 is audit-unusable. Waterfall-Llama is the only configuration retaining acceptable low-FPR performance under paraphrase (0.71).

F.1.4. Source Attribution

Blog attribution is noisier than C4 ($K = 20$ vs. $K = 10$ at weaker per-row signal). Waterfall stays perfect on both bases; SynthID reaches 1.0 on Llama-8B (pooling sharpens noisy per-row scores). Unigram is the worst at 0.700 on both bases.

Model	Base	No Watermark	Δ	KGW	Δ	Unigram	Δ	Waterfall	Δ	SynthID	Δ	EXP-Gumbel	Δ
Llama-8B	0.690	0.663	−0.028	0.638	−0.024	0.641	−0.022	**0.649**	**−0.013**	0.639	−0.024	0.643	−0.020
Qwen-7B	0.708	0.684	−0.024	0.663	−0.021	0.661	−0.023	**0.666**	**−0.018**	0.664	−0.020	0.658	−0.026
Qwen-14B	0.761	0.744	−0.017	**0.725**	**−0.020**	0.716	−0.028	0.722	−0.023	0.716	−0.028	0.719	−0.025
Gemma-9B	0.731	0.689	−0.042	0.653	−0.036	**0.659**	**−0.029**	0.654	−0.035	0.650	−0.039	0.651	−0.038

† No Watermark Δ is relative to Base; all other Δ are relative to No Watermark.

Table 21. Average zero-shot accuracy across six LM-Eval-Harness tasks (ARC-Easy, ARC-Challenge, HellaSwag, WinoGrande, TruthfulQA-MC2, MMLU) on the C4 realnewslike dataset. Bold indicates the least degradation among watermarks.

Watermark	Model	Before Attack	After Attack		
			Tok. Drop	Syn. Sub.	Paraphrase
Waterfall	Llama-8B	**1.0000**	**1.0000**	**1.0000**	**0.9610**
	Gemma-9B	**1.0000**	0.9995	0.9993	0.8094
KGW	Llama-8B	**1.0000**	0.9998	0.9998	0.8871
	Gemma-9B	0.9986	0.9895	0.9875	0.7050
Unigram	Llama-8B	0.9878	0.9821	0.9798	0.8159
	Gemma-9B	0.9654	0.9412	0.9407	0.7579
SynthID	Llama-8B	0.9965	0.9780	0.9847	0.6708
	Gemma-9B	0.9817	0.9034	0.9068	0.5743
EXP-Gumbel	Llama-8B	0.9940	0.9864	0.9845	0.7751
	Gemma-9B	0.9863	0.9468	0.9541	0.6495

Table 22. Data-side detector AUROC on the Blog Authorship Corpus before and after attacks. Paraphrase reports AUROC under the Llama-3.1-8B-Instruct paraphraser.

F.1.5. Impact on Model Utility

Cost-of-fine-tuning (Base → No Watermark) is -0.024 to -0.071 (Gemma-9B largest); additional cost-of-watermarking (No Watermark → best) is -0.013 to -0.029, comparable to C4. Waterfall wins on Llama-8B / Qwen-14B; Unigram on Qwen-7B; KGW/EXP-Gumbel tied on Gemma-9B. SynthID is consistently the weakest utility watermark on Blog.

F.1.6. Machine Unlearning Verification

On the forget side, GD-200 lands within ± 0.02 of oracle on $8/10$ cells (effective unlearning across watermark families); NPO $\beta = 0.1$ stays near pre-unlearning (ineffective at this budget). On the retain side, all three operators stay near oracle on most cells; largest collateral damage is Unigram-Llama (GD-200 retain 0.662 vs. oracle 0.753). Two noise-floor cells (SynthID-Llama, Waterfall-Gemma) read as "no measurable per-key signal" on both sides — the metric correctly registers no signal rather than producing spurious confident readings. Unigram-Gemma NPO forget is missing pending a rerun.

F.2. MBPP/MBJSP

MBPP (Python) and MBJSP (JavaScript) (Austin et al., 2021). Both have a single data owner per release: the audit reduces to single-key data-side detection plus a $K = 1$ model-side audit. We report both to characterise how watermarking transfers to a low-entropy, syntactically-constrained surface form.

F.2.1. Effectiveness of Watermark on Corpus

Detection and Robustness to Attacks.

Code is harder than natural language: clean AUROC drops to 0.71–0.98 on MBPP and 0.73–0.97 on MBJSP, with SynthID falling to 0.71–0.78 (its non-distortionary signal cannot easily be learned in code's low-entropy token stream). Token drop / synonym substitution cost 0.05–0.20 (vs. 0.00–0.04 on text) since edits in code are far more disruptive. Llama-paraphrase drops AUROC to 0.40–0.84 on MBJSP, harder than MBPP because JavaScript has more idiomatic variation. EXP-Gumbel-Llama is strongest on MBPP (0.985); KGW-Llama on MBJSP (0.975); SynthID is weakest on both regardless

			LLM Judge			
Watermark	**Model**	**STS**	**Faithful (%)**	**Partial (%)**	**Unfaithful (%)**	**Soft Fidelity**
Waterfall	Llama-8B	**0.864**	44.0	**55.7**	0.3	0.718
	Gemma-9B	0.852	51.7	46.7	1.7	0.750
KGW	Llama-8B	0.839	52.3	47.0	0.7	0.758
	Gemma-9B	0.828	56.7	42.7	0.7	0.780
Unigram	Llama-8B	0.832	52.3	46.3	1.3	0.755
	Gemma-9B	0.825	56.7	43.0	0.3	0.782
SynthID	Llama-8B	0.834	47.0	52.0	1.0	0.730
	Gemma-9B	0.833	**58.0**	41.0	1.0	**0.785**
EXP-Gumbel	Llama-8B	0.832	42.7	56.3	1.0	0.708
	Gemma-9B	0.833	54.0	45.0	1.0	0.765

Table 23. Fidelity metrics (STS, LLM Judge) for the Blog Authorship Corpus. Lower Unfaithful is better; higher STS / Faithful / Soft Fidelity is better.

Watermark	**Model**	**AUROC**
Waterfall	Llama-8B	**1.000**
	Gemma-9B	**1.000**
KGW	Llama-8B	0.994
	Gemma-9B	0.982
Unigram	Llama-8B	0.998
	Gemma-9B	0.991
SynthID	Llama-8B	0.997
	Gemma-9B	0.968
EXP-Gumbel	Llama-8B	0.996
	Gemma-9B	0.985

Table 24. AUROC for per-id separability on the Blog Authorship Corpus.

of base.

Tables 38, 39, 40, 41, 42, and 43 report the corresponding TPR at fixed FPR operating points 1%, 5%, and 10% for each code dataset.

EXP-Gumbel-Llama on MBPP reaches TPR@10%FPR $= 0.956$ (the highest code-domain cell); Unigram-Llama on MBJSP retains TPR@10% $= 0.65$ even under LLM-paraphrase (the largest paraphrase-robustness margin we observe). Most other cells fall to TPR@1% ≤ 0.10 under paraphrase — code-domain watermarking is operationally fragile under realistic adversaries.

Does Watermarking Degrade Text Quality?

STS is 0.78–0.92 on MBPP and 0.65–0.91 on MBJSP. The LLM-judge columns reveal a sharp watermark-family effect absent on natural-language text: **SynthID degrades code semantics catastrophically**, with 83% unfaithful on MBPP-Llama and 85% on MBJSP-Llama (vs. $< 3\%$ on C4 / Blog). The remaining four watermarks preserve 84–96% of MBPP samples as faithful or partial; on MBJSP this drops to 77–84% with unfaithful rates 13–23%. Soft fidelity peaks at Waterfall-Gemma MBPP (0.858); SynthID sits at 0.11–0.26 on both code datasets. Code-domain deployment recommendation: choose KGW, Waterfall, Unigram, or EXP-Gumbel; SynthID is unsuitable.

F.2.2. Watermark Persistence through Training

Tables 47, 48, and 49 report the corresponding model-side TPR at fixed FPR operating points for MBPP.

Tables 51, 52, and 53 report the corresponding model-side TPR at fixed FPR operating points for MBJSP.

KGW and Waterfall preserve strong signal (AUROC 0.74–0.97); SynthID stays near chance on Llama-8B (0.51–0.54).

Watermark	Model	AUROC
Waterfall	Llama-8B	0.913
	Gemma-9B	0.922
KGW	Llama-8B	0.889
	Gemma-9B	0.935
Unigram	Llama-8B	0.853
	Gemma-9B	0.806
SynthID	Llama-8B	0.641
	Gemma-9B	0.731
EXP-Gumbel	Llama-8B	0.758
	Gemma-9B	0.845

Table 25. Model-side detection AUROC on Blog Authorship Corpus with $K=20$ candidate keys.

Watermark	Model	TPR@1%FPR
Waterfall	Llama-8B	0.340
	Gemma-9B	0.663
KGW	Llama-8B	0.207
	Gemma-9B	0.650
Unigram	Llama-8B	0.210
	Gemma-9B	0.400
SynthID	Llama-8B	0.047
	Gemma-9B	0.320
EXP-Gumbel	Llama-8B	0.130
	Gemma-9B	0.537

Table 26. Model-side detection TPR@1%FPR on the Blog Authorship Corpus dataset with $K=20$ candidate keys.

Waterfall and Unigram on Gemma-9B reach 0.94–0.97 on both code datasets, exceeding their Llama-8B counterparts despite Gemma being a slightly weaker substrate on natural-language text.

Watermark	Model	TPR@5%FPR
Waterfall	Llama-8B	0.663
	Gemma-9B	0.767
KGW	Llama-8B	0.533
	Gemma-9B	0.837
Unigram	Llama-8B	0.517
	Gemma-9B	0.610
SynthID	Llama-8B	0.227
	Gemma-9B	0.457
EXP-Gumbel	Llama-8B	0.283
	Gemma-9B	0.640

Table 27. Model-side detection TPR@5%FPR on the Blog Authorship Corpus dataset with $K{=}20$ candidate keys.

Watermark	Model	TPR@10%FPR
Waterfall	Llama-8B	0.750
	Gemma-9B	0.800
KGW	Llama-8B	0.713
	Gemma-9B	0.857
Unigram	Llama-8B	0.633
	Gemma-9B	0.670
SynthID	Llama-8B	0.327
	Gemma-9B	0.553
EXP-Gumbel	Llama-8B	0.403
	Gemma-9B	0.690

Table 28. Model-side detection TPR@10%FPR on the Blog Authorship Corpus dataset with $K{=}20$ candidate keys.

Watermark	Model	Before Attack	After Attack		
			Tok. Drop	Syn. Sub.	Paraphrase
Waterfall	Llama-8B	**1.0000**	**1.0000**	**1.0000**	**0.7133**
	Gemma-9B	**1.0000**	0.9860	0.9860	0.2400
KGW	Llama-8B	0.9980	0.9960	0.9960	0.4667
	Gemma-9B	0.9820	0.8400	0.8380	0.1600
Unigram	Llama-8B	0.8120	0.7400	0.7180	0.2433
	Gemma-9B	0.4940	0.3360	0.3200	0.0467
SynthID	Llama-8B	0.9500	0.7520	0.7860	0.1067
	Gemma-9B	0.8680	0.5240	0.5320	0.1067
EXP-Gumbel	Llama-8B	0.9560	0.8920	0.8960	0.2033
	Gemma-9B	0.8620	0.6000	0.6180	0.0700

Table 29. Data-side detector TPR@1%FPR on the Blog Authorship Corpus before and after attacks. Paraphrase reports TPR under the Llama-3.1-8B-Instruct paraphraser.

Watermark	Model	Before Attack	After Attack		
			Tok. Drop	Syn. Sub.	Paraphrase
Waterfall	Llama-8B	**1.0000**	**1.0000**	**1.0000**	**0.8300**
	Gemma-9B	**1.0000**	0.9980	0.9960	0.3767
KGW	Llama-8B	**1.0000**	**1.0000**	**1.0000**	0.6067
	Gemma-9B	0.9960	0.9440	0.9440	0.2600
Unigram	Llama-8B	0.9360	0.9080	0.8940	0.3400
	Gemma-9B	0.8080	0.7020	0.6920	0.1600
SynthID	Llama-8B	0.9880	0.9140	0.9420	0.2667
	Gemma-9B	0.9360	0.7140	0.7280	0.2033
EXP-Gumbel	Llama-8B	0.9860	0.9580	0.9520	0.3333
	Gemma-9B	0.9380	0.7640	0.8000	0.1800

Table 30. Data-side detector TPR@5%FPR on the Blog Authorship Corpus before and after attacks. Paraphrase reports TPR under the Llama-3.1-8B-Instruct paraphraser.

Watermark	Model	Before Attack	After Attack		
			Tok. Drop	Syn. Sub.	Paraphrase
Waterfall	Llama-8B	**1.0000**	**1.0000**	**1.0000**	**0.8867**
	Gemma-9B	**1.0000**	**1.0000**	**1.0000**	0.5767
KGW	Llama-8B	**1.0000**	**1.0000**	**1.0000**	0.6900
	Gemma-9B	0.9960	0.9760	0.9680	0.3467
Unigram	Llama-8B	0.9700	0.9620	0.9480	0.5200
	Gemma-9B	0.9360	0.8680	0.8800	0.3933
SynthID	Llama-8B	0.9940	0.9420	0.9700	0.3567
	Gemma-9B	0.9500	0.7700	0.7800	0.2433
EXP-Gumbel	Llama-8B	0.9920	0.9840	0.9760	0.4700
	Gemma-9B	0.9680	0.8500	0.8680	0.2233

Table 31. Data-side detector TPR@10%FPR on the Blog Authorship Corpus before and after attacks. Paraphrase reports TPR under the Llama-3.1-8B-Instruct paraphraser.

Watermark	Model	Attribution Accuracy
Waterfall	Llama-8B	**1.000**
	Gemma-9B	**1.000**
KGW	Llama-8B	0.900
	Gemma-9B	0.950
Unigram	Llama-8B	0.700
	Gemma-9B	0.700
SynthID	Llama-8B	**1.000**
	Gemma-9B	0.950
EXP-Gumbel	Llama-8B	0.850
	Gemma-9B	**1.000**

Table 32. Pooled attribution accuracy on the Blog Authorship Corpus ($K = 20$ candidate keys). Each true-key group's per-row scores are averaged before the argmax over candidate keys.

		No Watermark		KGW		Unigram		Waterfall		SynthID		EXP-Gumbel	
Model	**Base**		Δ		Δ		Δ		Δ		Δ		Δ
Llama-8B	0.690	0.652	−0.038	0.624	−0.029	0.625	−0.027	**0.632**	**−0.020**	0.616	−0.036	0.622	−0.030
Qwen-7B	0.708	0.677	−0.031	0.645	−0.032	**0.656**	**−0.021**	0.648	−0.028	0.642	−0.035	0.647	−0.030
Qwen-14B	0.761	0.731	−0.030	0.699	−0.033	0.698	−0.033	**0.703**	**−0.028**	0.696	−0.035	0.696	−0.035
Gemma-9B	0.731	0.660	−0.071	**0.647**	**−0.014**	0.643	−0.017	0.643	−0.017	0.642	−0.018	**0.646**	**−0.014**

† No Watermark Δ is relative to Base; all other Δ are relative to No Watermark.

Table 33. Average zero-shot accuracy across six LM-Eval-Harness tasks (ARC-Easy, ARC-Challenge, HellaSwag, WinoGrande, TruthfulQA-MC2, MMLU) on the Blog Authorship Corpus.

Watermark	**Model**	**Pre**	**GD-200**	**NPO** β=0.1	**Oracle**
Waterfall	Llama-8B	0.654	**0.448**	0.654	0.490
	Gemma-9B	0.553	0.569	0.572	0.427
KGW	Llama-8B	0.616	**0.481**	0.625	0.477
	Gemma-9B	0.655	**0.489**	0.637	0.464
Unigram	Llama-8B	0.734	**0.471**	0.731	0.513
	Gemma-9B	0.705	0.639	—	0.510
SynthID	Llama-8B	0.524	**0.492**	0.548	0.568
	Gemma-9B	0.439	**0.422**	**0.403**	0.437
EXP-Gumbel	Llama-8B	0.497	**0.437**	**0.451**	0.467
	Gemma-9B	0.526	**0.440**	0.536	0.451

Table 34. Machine unlearning verification on the Blog Authorship Corpus, forget-side model-side detection AUROC. Pre = full-FT before unlearning; GD-200 = gradient-difference at 200 steps; NPO $\beta = 0.1$ at 500 steps; Oracle = retrain-on-retain reference.

Watermark	**Model**	**Pre**	**GD-200**	**NPO** β=0.1	**Oracle**
Waterfall	Llama-8B	0.805	0.722	**0.789**	0.807
	Gemma-9B	0.708	**0.680**	**0.720**	0.731
KGW	Llama-8B	0.745	0.652	**0.755**	0.753
	Gemma-9B	0.729	**0.691**	**0.737**	0.729
Unigram	Llama-8B	0.747	0.662	**0.756**	0.753
	Gemma-9B	0.697	**0.677**	**0.667**	0.682
SynthID	Llama-8B	**0.512**	**0.521**	**0.533**	0.494
	Gemma-9B	**0.531**	**0.515**	0.585	0.522
EXP-Gumbel	Llama-8B	0.570	**0.509**	**0.575**	0.594
	Gemma-9B	0.570	0.547	**0.562**	0.592

Table 35. Machine unlearning verification on the Blog Authorship Corpus, retain-side model-side detection AUROC. Same column structure as Table 34.

Watermark	Model	Before Attack	After Attack		
			Tok. Drop	Syn. Sub.	Paraphrase
Waterfall	Llama-8B	0.9488	0.9226	0.9186	0.7035
	Gemma-9B	0.9054	0.7891	0.7874	0.7281
KGW	Llama-8B	0.9615	0.9425	0.9408	0.6966
	Gemma-9B	0.9414	0.9050	0.9078	0.8020
Unigram	Llama-8B	0.8527	0.8079	0.8130	0.7579
	Gemma-9B	0.8039	0.5968	0.6024	0.6992
SynthID	Llama-8B	0.7070	0.5952	0.6116	0.5481
	Gemma-9B	0.7786	0.6412	0.6456	0.5340
EXP-Gumbel	Llama-8B	**0.9847**	**0.9495**	**0.9570**	0.7381
	Gemma-9B	0.8877	0.7685	0.7819	0.5932

Table 36. Data-side detector AUROC on the MBPP (Python code) dataset before and after attacks. Paraphrase reports AUROC under the Llama-3.1-8B-Instruct paraphraser.

Watermark	Model	Before Attack	After Attack		
			Tok. Drop	Syn. Sub.	Paraphrase
Waterfall	Llama-8B	0.9288	0.8542	0.8477	0.4040
	Gemma-9B	0.7763	0.6633	0.6570	0.3979
KGW	Llama-8B	**0.9747**	**0.9248**	**0.9257**	0.7001
	Gemma-9B	0.8547	0.8348	0.8425	0.5523
Unigram	Llama-8B	0.8974	0.8084	0.8088	**0.8396**
	Gemma-9B	0.8445	0.7190	0.7290	0.6801
SynthID	Llama-8B	0.7270	0.6486	0.6672	0.5414
	Gemma-9B	0.7641	0.5967	0.6102	0.5321
EXP-Gumbel	Llama-8B	0.9645	0.8756	0.9056	0.6423
	Gemma-9B	0.8967	0.6949	0.7067	0.5062

Table 37. Data-side detector AUROC on the MBJSP (JavaScript code) dataset before and after attacks. Paraphrase reports AUROC under the Llama-3.1-8B-Instruct paraphraser.

Watermark	Model	Before Attack	After Attack		
			Tok. Drop	Syn. Sub.	Paraphrase
Waterfall	Llama-8B	0.6200	0.5180	0.4700	0.0967
	Gemma-9B	0.2240	0.1000	0.0680	0.0300
KGW	Llama-8B	0.4540	0.2520	0.2480	0.0233
	Gemma-9B	0.3080	0.1620	0.1860	0.0500
Unigram	Llama-8B	0.2840	0.1500	0.1520	0.0967
	Gemma-9B	0.1400	0.0280	0.0280	0.0667
SynthID	Llama-8B	0.2020	0.0780	0.0940	0.0667
	Gemma-9B	0.0440	0.0300	0.0260	0.0133
EXP-Gumbel	Llama-8B	**0.8400**	**0.5960**	**0.6280**	**0.1033**
	Gemma-9B	0.0960	0.0320	0.0340	0.0000

Table 38. Data-side detector TPR@1%FPR on the MBPP (Python code) dataset before and after attacks. Paraphrase reports TPR under the Llama-3.1-8B-Instruct paraphraser.

Watermark	Model	Before Attack	After Attack		
			Tok. Drop	Syn. Sub.	Paraphrase
Waterfall	Llama-8B	0.7540	0.6720	0.6420	0.2133
	Gemma-9B	0.5320	0.2760	0.2440	0.1200
KGW	Llama-8B	0.7920	0.6660	0.6400	0.0900
	Gemma-9B	0.6700	0.5180	0.5320	0.1833
Unigram	Llama-8B	0.5980	0.4240	0.4340	**0.3400**
	Gemma-9B	0.3080	0.0720	0.0780	0.1133
SynthID	Llama-8B	0.3380	0.1580	0.1640	0.1367
	Gemma-9B	0.2380	0.1060	0.0860	0.0500
EXP-Gumbel	Llama-8B	**0.9240**	**0.7820**	**0.8040**	0.2567
	Gemma-9B	0.4300	0.1920	0.2060	0.0767

Table 39. Data-side detector TPR@5%FPR on the MBPP (Python code) dataset before and after attacks. Paraphrase reports TPR under the Llama-3.1-8B-Instruct paraphraser.

Watermark	Model	Before Attack	After Attack		
			Tok. Drop	Syn. Sub.	Paraphrase
Waterfall	Llama-8B	0.8440	0.7780	0.7780	0.2967
	Gemma-9B	0.6820	0.3860	0.3640	0.2633
KGW	Llama-8B	0.8980	0.8100	0.8300	0.1667
	Gemma-9B	0.8520	0.7180	0.7140	0.3633
Unigram	Llama-8B	0.6640	0.5180	0.5460	**0.4267**
	Gemma-9B	0.4400	0.1300	0.1400	0.1867
SynthID	Llama-8B	0.4140	0.2280	0.2400	0.1833
	Gemma-9B	0.3640	0.1620	0.1600	0.0900
EXP-Gumbel	Llama-8B	**0.9560**	**0.8460**	**0.8820**	0.3900
	Gemma-9B	0.6380	0.3820	0.4040	0.1500

Table 40. Data-side detector TPR@10%FPR on the MBPP (Python code) dataset before and after attacks. Paraphrase reports TPR under the Llama-3.1-8B-Instruct paraphraser.

Watermark	Model	Before Attack	After Attack		
			Tok. Drop	Syn. Sub.	Paraphrase
Waterfall	Llama-8B	0.4200	0.2060	0.1700	0.0267
	Gemma-9B	0.1640	0.0980	0.0960	0.0167
KGW	Llama-8B	0.6180	0.3540	0.3500	0.0567
	Gemma-9B	0.0620	0.0500	0.0500	0.0000
Unigram	Llama-8B	0.6300	**0.3620**	0.3340	**0.4433**
	Gemma-9B	0.0560	0.0140	0.0100	0.0133
SynthID	Llama-8B	0.3380	0.2120	0.2280	0.0800
	Gemma-9B	0.1460	0.0460	0.0460	0.0600
EXP-Gumbel	Llama-8B	**0.6400**	0.2820	**0.3540**	0.0533
	Gemma-9B	0.2080	0.0720	0.0660	0.0067

Table 41. Data-side detector TPR@1%FPR on the MBJSP (JavaScript code) dataset before and after attacks. Paraphrase reports TPR under the Llama-3.1-8B-Instruct paraphraser.

Watermark	Model	Before Attack	After Attack		
			Tok. Drop	Syn. Sub.	Paraphrase
Waterfall	Llama-8B	0.6860	0.4680	0.4380	0.0633
	Gemma-9B	0.3620	0.2300	0.2040	0.0400
KGW	Llama-8B	**0.8820**	**0.6320**	**0.6200**	0.2000
	Gemma-9B	0.3080	0.2860	0.3120	0.0367
Unigram	Llama-8B	0.7180	0.5280	0.5280	**0.5567**
	Gemma-9B	0.2260	0.0660	0.0620	0.0433
SynthID	Llama-8B	0.4460	0.3000	0.3140	0.1600
	Gemma-9B	0.3220	0.1480	0.1580	0.1167
EXP-Gumbel	Llama-8B	0.8280	0.5360	0.5840	0.1433
	Gemma-9B	0.6000	0.2560	0.2440	0.0800

Table 42. Data-side detector TPR@5%FPR on the MBJSP (JavaScript code) dataset before and after attacks. Paraphrase reports TPR under the Llama-3.1-8B-Instruct paraphraser.

Watermark	Model	Before Attack	After Attack		
			Tok. Drop	Syn. Sub.	Paraphrase
Waterfall	Llama-8B	0.7940	0.5880	0.5780	0.0933
	Gemma-9B	0.4620	0.2980	0.2740	0.0633
KGW	Llama-8B	**0.9380**	**0.7640**	**0.7920**	0.3033
	Gemma-9B	0.4940	0.4540	0.4960	0.1100
Unigram	Llama-8B	0.7720	0.6000	0.6180	**0.6533**
	Gemma-9B	0.3960	0.1320	0.1500	0.0767
SynthID	Llama-8B	0.4960	0.3600	0.3700	0.2067
	Gemma-9B	0.4700	0.2560	0.2660	0.1700
EXP-Gumbel	Llama-8B	0.9000	0.6560	0.7240	0.2633
	Gemma-9B	0.7120	0.3300	0.3440	0.1067

Table 43. Data-side detector TPR@10%FPR on the MBJSP (JavaScript code) dataset before and after attacks. Paraphrase reports TPR under the Llama-3.1-8B-Instruct paraphraser.

			LLM Judge			
Watermark	Model	STS	Faithful (%)	Partial (%)	Unfaithful (%)	Soft Fidelity
Waterfall	Llama-8B	**0.917**	62.3	26.7	11.0	0.757
	Gemma-9B	0.917	**76.0**	19.7	4.3	**0.858**
KGW	Llama-8B	0.841	39.0	**55.7**	5.3	0.668
	Gemma-9B	0.850	67.7	26.0	6.3	0.807
Unigram	Llama-8B	0.876	56.3	37.0	6.7	0.748
	Gemma-9B	0.872	71.0	24.3	4.7	0.832
SynthID	Llama-8B	0.777	4.7	12.3	83.0	0.108
	Gemma-9B	0.851	6.3	31.7	62.0	0.222
EXP-Gumbel	Llama-8B	0.857	52.0	40.0	8.0	0.720
	Gemma-9B	0.877	71.3	25.3	3.3	0.840

Table 44. Fidelity metrics (STS, LLM Judge) for the MBPP (Python code) dataset. Lower Unfaithful is better; higher STS / Faithful / Soft Fidelity is better.

			LLM Judge			
Watermark	**Model**	**STS**	**Faithful (%)**	**Partial (%)**	**Unfaithful (%)**	**Soft Fidelity**
Waterfall	Llama-8B	**0.906**	43.7	39.3	17.0	0.633
	Gemma-9B	0.809	57.3	22.7	20.0	0.687
KGW	Llama-8B	0.872	37.7	**48.7**	13.7	0.620
	Gemma-9B	0.747	52.3	25.0	22.7	0.648
Unigram	Llama-8B	0.851	45.3	38.3	16.3	0.645
	Gemma-9B	0.777	**61.7**	19.3	19.0	**0.713**
SynthID	Llama-8B	0.652	10.3	4.3	85.3	0.125
	Gemma-9B	0.689	18.7	14.3	67.0	0.258
EXP-Gumbel	Llama-8B	0.890	47.3	35.3	17.3	0.650
	Gemma-9B	0.761	50.3	27.3	22.3	0.640

Table 45. Fidelity metrics (STS, LLM Judge) for the MBJSP (JavaScript code) dataset. Lower Unfaithful is better; higher STS / Faithful / Soft Fidelity is better.

Watermark	**Model**	**AUROC**
Waterfall	Llama-8B	0.797
	Gemma-9B	0.940
KGW	Llama-8B	0.882
	Gemma-9B	0.813
Unigram	Llama-8B	0.718
	Gemma-9B	0.928
SynthID	Llama-8B	0.509
	Gemma-9B	0.670
EXP-Gumbel	Llama-8B	0.778
	Gemma-9B	0.705

Table 46. Model-side detection AUROC on MBPP (Python) with $K=1$ (single owner).

Watermark	**Model**	**TPR@1%FPR**
Waterfall	Llama-8B	0.055
	Gemma-9B	0.150
KGW	Llama-8B	0.400
	Gemma-9B	0.250
Unigram	Llama-8B	0.170
	Gemma-9B	0.470
SynthID	Llama-8B	0.013
	Gemma-9B	0.013
EXP-Gumbel	Llama-8B	0.077
	Gemma-9B	0.040

Table 47. Model-side detection TPR@1%FPR on the MBPP (Python code) dataset with $K=1$ candidate keys.

Watermark	Model	TPR@5%FPR
Waterfall	Llama-8B	0.265
	Gemma-9B	0.763
KGW	Llama-8B	0.585
	Gemma-9B	0.500
Unigram	Llama-8B	0.335
	Gemma-9B	0.780
SynthID	Llama-8B	0.107
	Gemma-9B	0.160
EXP-Gumbel	Llama-8B	0.353
	Gemma-9B	0.217

Table 48. Model-side detection TPR@5%FPR on the MBPP (Python code) dataset with $K=1$ candidate keys.

Watermark	Model	TPR@10%FPR
Waterfall	Llama-8B	0.425
	Gemma-9B	0.893
KGW	Llama-8B	0.675
	Gemma-9B	0.603
Unigram	Llama-8B	0.405
	Gemma-9B	0.853
SynthID	Llama-8B	0.197
	Gemma-9B	0.250
EXP-Gumbel	Llama-8B	0.530
	Gemma-9B	0.360

Table 49. Model-side detection TPR@10%FPR on the MBPP (Python code) dataset with $K=1$ candidate keys.

Watermark	Model	AUROC
Waterfall	Llama-8B	0.739
	Gemma-9B	**0.965**
KGW	Llama-8B	0.830
	Gemma-9B	0.764
Unigram	Llama-8B	0.698
	Gemma-9B	**0.965**
SynthID	Llama-8B	0.540
	Gemma-9B	0.584
EXP-Gumbel	Llama-8B	0.752
	Gemma-9B	0.764

Table 50. Model-side detection AUROC on MBJSP (JavaScript) with $K=1$ (single owner).

Watermark	Model	TPR@1%FPR
Waterfall	Llama-8B	0.030
	Gemma-9B	0.713
KGW	Llama-8B	0.220
	Gemma-9B	0.263
Unigram	Llama-8B	0.285
	Gemma-9B	0.767
SynthID	Llama-8B	0.010
	Gemma-9B	0.010
EXP-Gumbel	Llama-8B	0.080
	Gemma-9B	0.260

Table 51. Model-side detection TPR@1%FPR on the MBJSP (JavaScript code) dataset with $K=1$ candidate keys.

Watermark	Model	TPR@5%FPR
Waterfall	Llama-8B	0.225
	Gemma-9B	0.887
KGW	Llama-8B	0.510
	Gemma-9B	0.387
Unigram	Llama-8B	0.415
	Gemma-9B	0.890
SynthID	Llama-8B	0.057
	Gemma-9B	0.070
EXP-Gumbel	Llama-8B	0.217
	Gemma-9B	0.397

Table 52. Model-side detection TPR@5%FPR on the MBJSP (JavaScript code) dataset with $K=1$ candidate keys.

Watermark	Model	TPR@10%FPR
Waterfall	Llama-8B	0.330
	Gemma-9B	0.913
KGW	Llama-8B	0.635
	Gemma-9B	0.487
Unigram	Llama-8B	0.503
	Gemma-9B	0.930
SynthID	Llama-8B	0.120
	Gemma-9B	0.147
EXP-Gumbel	Llama-8B	0.373
	Gemma-9B	0.490

Table 53. Model-side detection TPR@10%FPR on the MBJSP (JavaScript code) dataset with $K=1$ candidate keys.