REBOUND: Radioactive Elasticity-Based Objectives for Unwatermarking Neural Decoders

Arul Kolla

Introduction

Modern machine learning systems are increasingly trained on large, partially proprietary corpora. Data owners therefore seek mechanisms to verify whether their data contributed to a model's training procedure.[1] *Watermarking* addresses this by embedding label-preserving perturbations into training examples, creating a statistically detectable signature in the resulting model.[2] This phenomenon is often called *radioactivity*, since the model's outputs have a sort of "trace" that is useful for detection.

Parallel research frames model alignment through the lens of *data compression*. Ji et al. propose modeling datasets as forces acting on a spring within an abstract data space.[3] In this geometric view, model coordinates are defined by normalized compression rates. Pre-training creates a deep potential energy basin, while fine-tuning induces shallow displacements governed by a Hooke's-law-like relationship.[4]

These two perspectives naturally intersect. Radioactive watermarking is a special case of fine-tuning on a small, carefully engineered dataset: it pulls the model toward a direction that improves compression on a marked set of examples. Elasticity suggests that such displacements may be fragile under subsequent training. This motivates our central question:

[1] B. G. A. Tekgul and N. Asokan, "On the Effectiveness of Dataset Watermarking in Adversarial Settings," *Proceedings of the 2022 ACM on Asia Conference on Computer and Communications Security (ASIACCS)*, pp. 1243–1245, 2022.

[2] A. Sablayrolles, M. Douze, C. Schmid, and H. Jégou, "Radioactive data: Tracing through training," *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 2020.

[3] J. Ji, K. Wang, T. Qiu, et al., "Language Models Resist Alignment: Evidence From Data Compression," *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2025.

[4] J. Ji, K. Wang, T. Qiu, et al., "Language Models Resist Alignment: Evidence From Data Compression," *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2025.

> If radioactive watermarks behave like forces on a spring in data space, to what extent can an adversary apply a counter-force via fine-tuning to erase the radioactive signal without sacrificing downstream utility?

We analyze this question in a unified compression-based framework. Starting from a pre-trained model M_0, we obtain a watermarked model M_w by fine-tuning on a radioactive dataset D_w (or teacher outputs encoding radioactivity), which stretches the model along a dedicated axis associated with improved compression of radioactive examples. We then apply additional fine-tuning on a second dataset D_t designed as an *un-finetuning* force, with the goal of relaxing or reversing the displacement along the D_w axis while preserving performance on natural evaluation distributions.

Motivation

Dataset ownership and radioactive watermarking

Our starting point is the view that dataset watermarking is a tool for *ownership verification*: a data owner perturbs a subset of training examples so that any model trained on that data exhibits a detectable statistical signature under a secret test. We build on radioactive data as a canonical instantiation of this idea.[5]

Given a base dataset D_w, radioactive construction produces a perturbed $\tilde{D}_w$ by adding small, label-preserving signals in feature space, constrained to be imperceptible. Models trained on $\tilde{D}_w$ exhibit a consistent loss gap between radioactive and matched clean samples that can be turned into a powerful hypothesis test, even when only a small fraction of training points are radioactive.[6] Subsequent analysis shows that these signatures can survive realistic attacks such as model extraction, though detection degrades in low-data regimes and can behave counterintuitively across black-box vs. white-box settings.[7]

In this project, the central object is a language model M_w trained on a mixture of pre-training data and a radioactive dataset D_w, such that a trusted verifier can detect training on D_w, while an adversary may seek to erase this signature.

[5] A. Sablayrolles, M. Douze, C. Schmid, and H. Jégou, "Radioactive data: Tracing through training," *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 2020.

[6] A. Sablayrolles, M. Douze, C. Schmid, and H. Jégou, "Radioactive data: Tracing through training," *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 2020.

[7] B. G. A. Tekgul and N. Asokan, "On the Effectiveness of Dataset Watermarking in Adversarial Settings," *Proceedings of the 2022 ACM on Asia Conference on Computer and Communications Security (ASIACCS)*, pp. 1243–1245, 2022.

> **Note**
>
> Throughout, we treat a **radioactive dataset** D_w as a base dataset augmented with small, optimized perturbations such that: (1) perturbations are norm-bounded and label-preserving; (2) any model trained on D_w enjoys a statistically significant loss reduction on radioactive vs. matched clean samples; and (3) this loss reduction is robust across a specified family of architectures and training setups.[8]

Elasticity of alignment and compression-based metrics

Alignment methods, such as supervised instruction tuning and reinforcement learning from human feedback, are typically applied to a large pre-trained model M_0 using comparatively small curated datasets D_a.[9] Empirically, most of the model's factual knowledge and linguistic competence arises in pre-training, while alignment nudges the model toward preferred behaviors on user-facing tasks.[10] This suggests that alignment operates in a low-measure region of parameter space relative to pre-training.

> **Note**
>
> Throughout this work, we use the **normalized negative log-likelihood per token** as a compression metric. For a model M and dataset D, we define
>
> $$\gamma_D(M) = \frac{1}{|D|} \sum_{(x,y)\in D} \frac{1}{T} \sum_{t=1}^{T} (-\log p_M(y_t \mid x, y_{<t})),$$
>
> where $y = (y_1, \ldots, y_T)$ is the target sequence. Lower $\gamma_D(M)$ indicates that M assigns higher probability to D and better matches its empirical distribution.[11] By tracking $\gamma_{D_w}(M)$ and $\gamma_{D_t}(M)$, together with standard radioactive detection statistics, we treat watermarking and un-finetuning as **coupled forces** in data space and quantify how far the model can be shifted before task utility degrades.

To build intuition, we provide an interactive widget in Figure 1 that illustrates how compression rates respond to model bias.

Ji et al. formalize this intuition under the name **elasticity**, using normalized compression rates

[8] A. Sablayrolles, M. Douze, C. Schmid, and H. Jégou, "Radioactive data: Tracing through training," *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 2020.

[9] C. Zhou, P. Liu, P. Xu, et al., "LIMA: Less Is More for Alignment," *Advances in Neural Information Processing Systems (NeurIPS)*, Vol. 36, 2023.

[10] C. Zhou, P. Liu, P. Xu, et al., "LIMA: Less Is More for Alignment," *Advances in Neural Information Processing Systems (NeurIPS)*, Vol. 36, 2023.

[11] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed., Wiley, 2006.

Figure 1: This toy dataset has three kinds of tokens: “MIT”, “TIM”, and “other”. The true dataset distribution is fixed (roughly 50% MIT, 40% TIM, 10% other). Use the slider to bias the model toward MIT or TIM and watch how the average negative log-likelihood (compression rate) changes.

to measure how models respond to fine-tuning.[12] For a model M and dataset D, they define $\gamma_D(M)$ as above and consider compression advantages of the form $\Delta\gamma_D(M) = \gamma_D(M_0) - \gamma_D(M)$. Ji et al. show that small alignment datasets can substantially improve compression on alignment datasets but that these gains are fragile: subsequent fine-tuning on other data quickly erases them, especially in larger, heavily pre-trained models.[13]

Elasticity as a generic un-watermarking mechanism

We treat radioactive watermarking as an extreme instance of this asymmetry. Both watermarking and alignment are induced by comparatively small, specialized datasets:

- In the radioactive case, D_w consists of marked examples designed so that training pulls the model toward a narrow carrier subspace, producing a detectable loss gap.[14]
- In the alignment case, D_a shifts behavior on user-facing tasks away from the raw pre-training prior.[15]

From the compression perspective, a successful watermark enforces $\gamma_{D_w}(M_w) \ll \gamma_{D_w}(M_0)$, exactly analogous to the alignment objective on D_a. This motivates our central hypothesis that radioactive signatures should be *elastic* in the same sense as alignment.

! Main hypothesis (Elastic un-watermarking)

Let M_0 be a pre-trained language model and M_w a watermarked model obtained by fine-tuning M_0 on a radioactive dataset D_w. Assume that a verifier can detect training on D_w via an empirical compression advantage on D_w. Suppose we have only black-box or adapter-level access to M_w.
Then there exists a fine-tuning dataset D_t and training procedure $\mathcal{T}$, which are **agnostic** of D_w and the watermark key, such that $M_t := \mathcal{T}(M_w; D_t)$ satisfies $\gamma_{D_w}(M_t) \approx \gamma_{D_w}(M_0)$.

[12] J. Ji, K. Wang, T. Qiu, et al., “Language Models Resist Alignment: Evidence From Data Compression,” *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2025.
[13] J. Ji, K. Wang, T. Qiu, et al., “Language Models Resist Alignment: Evidence From Data Compression,” *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2025.
[14] A. Sablayrolles, M. Douze, C. Schmid, and H. Jégou, “Radioactive data: Tracing through training,” *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 2020.
[15] C. Zhou, P. Liu, P. Xu, et al., “LIMA: Less Is More for Alignment,” *Advances in Neural Information Processing Systems (NeurIPS)*, Vol. 36, 2023.

Our black-box adversary with fine-tuning capabilities

We consider an adversary who does not control the original training pipeline and only observes the released watermarked model M_w, as in black-box radioactive verification where evidence of watermarked training is obtained solely via queries.[16]

The adversary has black-box query access to M_w sufficient to obtain token-level probabilities or samples for arbitrary prompts, can perform parameter-efficient fine-tuning (e.g., LoRA or other adapter methods) on their own data,[17] and can evaluate candidate models on arbitrary datasets, including public benchmarks and proprietary task distributions. They do **not** have access to M_0, the watermarking key, the radioactive dataset D_w, or any clean-radioactive pairs.

Let Det denote the verifier's detection statistic. For radioactive data, Det is based on a loss comparison between radioactive and clean samples plus a hypothesis test at a chosen significance level.[18] The adversary seeks a model M_t such that $\mathsf{Det}(M_t) \approx \mathsf{Det}(M_{\text{clean}})$, where M_{clean} denotes typical unwatermarked models trained without D_w.

Because the adversary does not know Det exactly, they rely on *proxies* such as changes in compression rate $\gamma_D(M)$ on accessible datasets D and behavioral similarity between M_t and unwatermarked baselines. It is elasticity which links these proxies to training geometry: changes in $\gamma_D(M)$ across different D are driven by dataset size, gradient alignment, and pre-training scale.[19]

> **Note**
>
> **Leverage.** Even without access to M_0 or D_w, an adversary is still able to probe the loss landscape of M_w on large unlabeled corpora, identify high-loss regions, and use these as candidate directions for un-finetuning. Elasticity theory predicts when such directions will counteract watermark-induced displacements.

[16] A. Sablayrolles, M. Douze, C. Schmid, and H. Jégou, "Radioactive data: Tracing through training," *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 2020.

[17] Llama Team, AI @ Meta, "The Llama 3 Herd of Models," *arXiv preprint arXiv:2407.21783*, 2024.

[18] A. Sablayrolles, M. Douze, C. Schmid, and H. Jégou, "Radioactive data: Tracing through training," *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 2020.

[19] J. Ji, K. Wang, T. Qiu, et al., "Language Models Resist Alignment: Evidence From Data Compression," *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2025.

A compression view of elasticity and radioactive watermarks

This section develops the compression-based framework used to analyze elasticity and adapts it to radioactive watermarking for language models.[20] [21] [22] Let M_0 be a base model, M_w a radioactive model obtained by fine-tuning on D_w, and $M_{w\to t}$ the result of additional fine-tuning on D_t.

Datasets as forces in data space

We treat each dataset D as defining a coordinate axis in a data space. For a fixed base model M_0 and any model M, the *compression advantage* on D is

$$\Delta\gamma_D(M) = \gamma_D(M_0) - \gamma_D(M).$$

A positive $\Delta\gamma_D(M)$ indicates that M compresses D better than M_0.

For a collection of datasets $\{D_i\}_{i=1}^n$, the vector $(\Delta\gamma_{D_1}(M), \dots, \Delta\gamma_{D_n}(M))$ embeds each model as a point in $\mathbb{R}^n$. Ji et al. show that, for modest fine-tuning and small mixtures of datasets, these coordinates respond to new training data approximately linearly: changes in $\Delta\gamma_{D_i}$ are controlled by a matrix of *elasticity coefficients* E_{ij} that depend on gradient inner products at M_0.[23]

Intuitively, training on dataset D_j exerts a *force* that increases $\Delta\gamma_{D_j}$, while pre-training on D_p acts as a strong anchor keeping the system near M_0. The elastic response of the system determines how changes along one axis (e.g., D_t) propagate to others (e.g., D_w).

Figure 2: An interactive three-spring model of our setup.

Loss-based selection as a proxy for anti-watermark directions

In the black-box setting, the adversary cannot compute gradients or elasticity coefficients for the watermark dataset D_w directly. However, they can query the watermarked model M_w on a large candidate pool U and measure token-level losses.

[20] J. Ji, K. Wang, T. Qiu, et al., "Language Models Resist Alignment: Evidence From Data Compression," *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2025.

[21] A. Sablayrolles, M. Douze, C. Schmid, and H. Jégou, "Radioactive data: Tracing through training," *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 2020.

[22] T. Sander, P. Fernandez, A. Durmus, M. Douze, and T. Furon, "Watermarking Makes Language Models Radioactive," *Advances in Neural Information Processing Systems (NeurIPS)*, Vol. 37, 2024.

[23] J. Ji, K. Wang, T. Qiu, et al., "Language Models Resist Alignment: Evidence From Data Compression," *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2025.

Intuitively, examples with high loss under M_w are those on which the watermark-induced bias is most misaligned with the pre-training distribution. Selecting such examples for fine-tuning should therefore push the model back toward the pre-training optimum and reduce the compression advantage on D_w.

We now formalize this intuition in an idealized model.

Theorem 1

Consider a watermarked model with parameters $\theta_w \in \mathbb{R}^d$. We assume there is a *watermark direction* $v \in \mathbb{R}^d$ with unit norm such that the gradient of the compression rate on the watermark dataset D_w at θ_w points exactly along v: more precisely, $\nabla_\theta \gamma_{D_w}(\theta_w) = -\|g_w\| v$ for some scalar magnitude $\|g_w\| > 0$. In other words, moving in the direction $+v$ decreases the watermark strength (increases γ_{D_w}), and moving in the direction $-v$ strengthens it.
For each training example z in a large candidate pool U, let $G(z) = \nabla_\theta \ell(\theta_w; z)$ denote the per-example gradient at θ_w. We model these gradients as isotropic Gaussian noise: $G(z) \sim \mathcal{N}(0, \sigma^2 I_d)$ for some variance parameter $\sigma > 0$. The *loss score* we use to rank examples is the scalar projection of the gradient onto the watermark direction, $S(z) = v^\top G(z)$.
Fix a selection fraction $q \in (0, 1)$. The adversary constructs an un-finetuning dataset D_t by keeping exactly the top-q fraction of examples in U with the largest scores $S(z)$. Let G_t denote $G(z)$ when z is drawn from D_t. The adversary then performs infinitesimal gradient descent on D_t, so the parameter dynamics are $d\theta/d\tau = -\mathbb{E}[G_t]$.
Under these assumptions, it follows that the expected gradient on D_t is exactly aligned with the watermark direction v, with a positive scalar coefficient that depends only on q and the Gaussian geometry. Writing $\lambda(q) = \phi(\Phi^{-1}(1-q))/q$, where ϕ and Φ are the standard normal pdf and cdf, we have $\mathbb{E}[G_t] = \sigma\lambda(q)v$. As a consequence, the instantaneous rate of change of the compression advantage on D_w under this un-finetuning dynamics satisfies

$$\frac{d}{d\tau}\Delta\gamma_{D_w}(\theta(\tau))\Big|_{\tau=0} = -\|g_w\|\sigma\lambda(q) < 0.$$

Here $\Delta\gamma_{D_w}(\theta)$ is the compression advantage of the current model over a fixed clean baseline on D_w.

We provide a brief sketch of this below; the full proof appears in the Appendix.

Proof Sketch

Because (G, S) is jointly Gaussian and S is a one-dimensional projection of G, the conditional mean $\mathbb{E}[G \mid S]$ must lie exactly in the span of v.
Conditioning on the event "top-q by S" amounts to conditioning on a one-sided truncation of a Gaussian; using the closed-form expression for the mean of a truncated normal distribution, we obtain that $\mathbb{E}[G \mid z \in D_t]$ is equal to $\sigma\lambda(q)v$, whose scalar factor $\lambda(q)$

depends only on q.
Under gradient descent on D_t we have $d\theta/d\tau = -\mathbb{E}[G_t] = -\sigma\lambda(q)v$. Combining this with $\nabla_\theta \gamma_{D_w}(\theta_w) = -\|g_w\|v$ and applying the chain rule, we obtain $\frac{d}{d\tau}\gamma_{D_w}(\theta(\tau))\big|_{\tau=0} = \|g_w\|\sigma\lambda(q)$, so the compression advantage $\Delta\gamma_{D_w}(\theta) = \gamma_{D_w}(\theta_0) - \gamma_{D_w}(\theta)$ evolves as

$$\frac{d}{d\tau}\Delta\gamma_{D_w}(\theta(\tau))\Big|_{\tau=0} = -\|g_w\|\sigma\lambda(q),$$

which is strictly negative for any $q \in (0, 1)$.

This theorem formalizes the idea that **loss-based selection is an anti-watermark direction**. Even though the adversary never sees the watermark data or direction explicitly, selecting the top-q high-loss examples by the scalar score $S(z) = v^\top G(z)$ ensures that the *average* gradient they train on points precisely along v, with a computable positive coefficient $\sigma\lambda(q)$.

Because the watermark gradient $\nabla_\theta \gamma_{D_w}(\theta_w)$ points along $-v$, gradient descent on D_t moves the model in the opposite direction and reduces the watermark's compression advantage at a rate exactly proportional to $\lambda(q)$.

Methods & Experimental Design

We set M_0 to be Llama-3.1-8B.[24] All fine-tuning is performed using low-rank adaptation (LoRA), and is orchestrated through the Tinker framework.

Datasets and watermarking setup

Radioactive dataset D_w

To simulate a realistic watermarking scenario, we use a publicly available radioactive dataset derived from the Maryland n-gram corpus and use them as supervised pairs.[25]

From this file, we carve out two disjoint subsets:

- a *watermark training set*, comprising of D_w^{train} examples from the corpus;
- a *watermark evaluation set*, comprising of D_w^{eval} examples from the corpus.

[24] Llama Team, AI @ Meta, "The Llama 3 Herd of Models," *arXiv preprint arXiv:2407.21783*, 2024.
[25] A. Sablayrolles, M. Douze, C. Schmid, and H. Jégou, "Radioactive data: Tracing through training," *ICML*, 2020.

Generic instruction dataset D_t

To approximate everyday instruction-following data, we use the Alpaca-cleaned dataset.[26] Each example is converted into a user/assistant pair by concatenating the instruction and (if present) the input field into a single user prompt, with the output as the assistant response.

Two roles are assigned to this distribution:

- A *held-out evaluation set* comprising of D_t^{eval} examples from the corpus. This serves as a proxy for generic task performance and allows us to quantify any collateral damage from un-watermarking.
- A larger *candidate pool* comprising of U examples from the coprus, which acts as the search space from which we construct adversarial training sets D_t using loss-based selection.

Compression metrics and validation signals

We report three derived quantities: the *radioactive compression rate* $\gamma_{D_w}(M)$, the *generic compression rate* $\gamma_{D_t}(M)$, and the *compression advantage* relative to the base model, $\Delta\gamma_D(M) = \gamma_D(M_0) - \gamma_D(M)$, for both D_w and D_t.

In addition, we track standard training and validation NLL to monitor overfitting and optimization stability. These validation metrics are evaluated periodically by snapshotting the current LoRA weights, creating a temporary sampling client, and running the same weighted NLL computation as above. This uniform treatment means that any movement of the spring in data space is reflected consistently across training and evaluation.

Adversarial D_t construction via loss-based selection

For each candidate conversation (x, y) in U, we compute the per-token SFT loss under M_w:

$$\ell_{M_w}(x, y) = \frac{1}{T} \sum -\log p_{M_w}(y_t \mid x, y_{<t}).$$

This loss is fully observable to the attacker via log-probability queries and aligns with the training objective. We then rank candidates based on ℓ_{M_w} and define two selection strategies:

- **High-loss selection.** D_t consists of the top k examples with the largest loss. These examples are where M_w disagrees most strongly with the Alpaca distribution. If Alpaca is closer to the pre-training data than the radioactive distribution (a plausible assumption), pushing the model to fit these high-loss regions should counteract the distortion introduced by D_w.

[26] R. Taori, I. Gulrajani, T. Zhang, et al., "Alpaca: A Strong, Replicable Instruction-Following Model," *Stanford Center for Research on Foundation Models (CRFM)*, 2023.

- **Random selection.** D_t is a uniform random subset of U. This baseline approximates the generic continued pre-training regime commonly studied in elasticity work.

By comparing how each strategy changes $\Delta\gamma_{D_w}$ and $\Delta\gamma_{D_t}$, we can test whether adversarial high-loss examples induce a qualitatively different elastic response from the model.

Training Curriculum and Elasticity Probes

The training curriculum is organized into three conceptual stages that mirror the elasticity thought experiment.

Stage 0: Base model probe

Before any fine-tuning, we measure $\gamma_{D_w}(M_0)$ and $\gamma_{D_t}(M_0)$. These values define the "rest position" of the spring and are used to compute compression advantages for all subsequent models. No parameters are updated in this stage; it purely establishes a reference.

Stage 1: Radioactive fine-tuning on D_w

We then train a LoRA adapter on D_w^{train}, obtaining the watermarked model M_w. The training objective is standard supervised fine-tuning on assistant tokens, with a warmup-stable-decay learning rate schedule. After Stage 1, we re-measure γ_{D_w} and γ_{D_t} to quantify the watermark-induced displacement. The model at this point is treated as the *only* object the attacker can access in the subsequent black-box setting.

Stage 2: Loss-based Dt construction

With M_w fixed, we score the candidate pool U using the loss function above and select a training set D_t according to one of the strategies (high-loss or random). Importantly, this selection is done once per experiment and cached; the resulting D_t is serialized and reused for all later runs that share the same strategy and size. This makes comparisons across different learning rate schedules or numbers of Dt epochs more meaningful, since they share the same adversarial curriculum.

Stage 3: Adversarial Dt fine-tuning and elasticity measurement

Starting from M_w, we fine-tune on D_t using the same LoRA configuration and WSD schedule as in Stage 1. At regular intervals during this stage, we snapshot the model and perform a standardized elasticity probe:

1. Compute γ_{D_w} and γ_{D_t} under the current model.
2. Convert these into compression advantages relative to M_0.
3. Evaluate validation NLL on D_w^{eval} and D_t^{eval}.

The trajectories of $\Delta\gamma_{D_w}$ and $\Delta\gamma_{D_t}$ over Dt steps are then interpreted as the motion of the model under the adversarial force induced by D_t. In the ideal un-watermarking scenario, high-loss D_t would cause $\Delta\gamma_{D_w}$ to decrease while maintaining positive $\Delta\gamma_{D_t}$, indicating that the model is regaining "generic" behavior without catastrophic degradation.

You can click on the callout below to view more specifics of our setup.

Experimental Hyperparameters & Setup

Model and infrastructure

Component	Choice
Base model M_0	LLaMA-family 8B model[27]
Adaptation	LoRA (rank 32)
Framework	Tinker (training + logprobs)
Max sequence length	512 tokens

Datasets

Role	Source
D_w^{train}	Radioactive Maryland corpus[28]
D_w^{eval}	Same as above
D_t^{eval}	Alpaca-cleaned[29]
Candidate pool U	Alpaca-cleaned
Adversarial D_t	Subset of U

Hyperparameters

Quantity	Value / Description
Batch size N	16 conversations per step
Learning rate peak	2×10^{-5} (WSD plateau)
WSD warmup fraction	5% of steps
WSD decay fraction	10% of steps
Dw epochs	1
Dt epochs	1
Validation frequency	every 10 steps
Gamma evaluation samples	up to 64 examples per dataset per probe

Results

Elastic response under random vs loss-based D_t selection

To evaluate elastic un-watermarking, we fix the radioactive fine-tuning stage on D_w and vary only how we construct the adversarial dataset D_t. For each strategy, we start from the same watermarked model M_w and track compression advantages $\Delta\gamma_{D_w}(M)$ and $\Delta\gamma_{D_t}(M)$ as we fine-tune on D_t.

Figure 3 plots these trajectories for two settings on 25% of our dataset, with the lighter lines showing the true γ values and the bold lines showing the time-smoothed averages:

- *Random D_t:* D_t is a uniform subset of generic Alpaca-style instructions.

[29]Llama Team, AI @ Meta, "The Llama 3 Herd of Models," *arXiv preprint arXiv:2407.21783*, 2024.
[29]A. Sablayrolles, M. Douze, C. Schmid, and H. Jégou, "Radioactive data: Tracing through training," *ICML*, 2020.
[29]R. Taori, I. Gulrajani, T. Zhang, et al., "Alpaca: A Strong, Replicable Instruction-Following Model," *Stanford Center for Research on Foundation Models (CRFM)*, 2023.

- *High-loss D_t:* D_t is constructed from the same pool, but restricted to examples with the highest SFT loss under M_w.

The x-axis shows the number of D_t SFT steps; the y-axis shows the compression advantage $\Delta\gamma_D(M) = \gamma_D(M_0) - \gamma_D(M)$ in nats per token, for both D_w and D_t.

Figure 3: Elastic response of compression advantages under random vs high-loss D_t selection. Each curve shows $\Delta\gamma_D(M)$ as we fine-tune M_w on D_t.

In the *random* regime, the two curves for $\Delta\gamma_{D_w}$ and $\Delta\gamma_{D_t}$ remain roughly parallel. As training proceeds, both advantages either drift upward together or downward together. This indicates that generic continued pre-training mainly moves the model along a direction that is shared by D_w and D_t: we see modest joint improvements or degradations, but no clear "un-watermarking" effect that selectively targets D_w.

In contrast, in the *high-loss* regime, the behavior qualitatively changes. As we fine-tune on high-loss D_t:

- the $\Delta\gamma_{D_t}$ curve increases over training, meaning the model becomes better at compressing the difficult Alpaca examples it initially mis-modeled;

- the $\Delta\gamma_{D_w}$ curve *decreases* over the same steps, reducing the compression advantage on the radioactive dataset.

The two curves cross during training: beyond a certain point the advantage on D_t surpasses that on D_w, and the model's bias visibly shifts away from the radioactive direction. This crossing pattern is exactly the elastic behavior we seek: a force applied along high-loss generic

directions pulls the model back toward performance on D_t while partially relaxing the "spring" attached to D_w.

To verify our hypothesis, we run an ablation study on our method with several different sizes of datasets.

In Figure 4, the title denotes the percentage of the coropora we use for training and evaluation (given that they are disjoint and their sizes are in the ratio $4:1$). The same general pattern holds:

- The random trendlines in red are largely parallel and do not seem to follow any regular pattern (the variation is due to the randomness of the initial finetuning step).
- The high-loss trendlines in green show how our model slowly gets worse at D_w and better at D_t.

As shown in the table below, while the random method keeps our compressions largely constant, the high loss method sharply decreases our compression on D_w.

Dataset %	Δ random_dt (smooth)	Δ random_dw (smooth)	Δ highloss_dt (smooth)	Δ highloss_dw (smooth)
30%	-0.041107	0.030222	0.029342	-0.051357
40%	0.207246	-0.011973	-0.041447	-0.110712
50%	-0.191412	-0.031874	-0.084028	-0.001198
60%	0.007150	0.079317	0.355340	-0.176700
70%	-0.048735	0.036518	-0.056934	-0.378964
80%	0.071952	0.020051	0.090087	-0.002440
90%	0.050559	-0.004245	0.124707	0.031995
100%	0.002387	0.054217	0.061912	-0.054670
Mean	**0.007255**	**0.021529**	**0.059872**	**-0.093006**

Implications for elastic un-watermarking

These results support the core hypothesis that watermarking behaves like a narrow alignment signal rather than a fundamentally rigid modification of the model. Random D_t primarily strengthens or weakens both D_w and D_t together, consistent with generic continued pre-training. High-loss D_t, however, induces an anisotropic response: it improves compression on D_t while degrading compression on D_w, even though the attacker never sees the watermark key or the clean base model M_0.

Quantitatively, the effect is partial rather than total: the final $\Delta\gamma_{D_w}$ remains above zero, so the watermark is weakened but not fully erased in this setting. Qualitatively, the divergence between random and high-loss trajectories demonstrates that the choice of adversarial data is critical: only carefully targeted high-loss examples exploit elasticity strongly enough to "bend

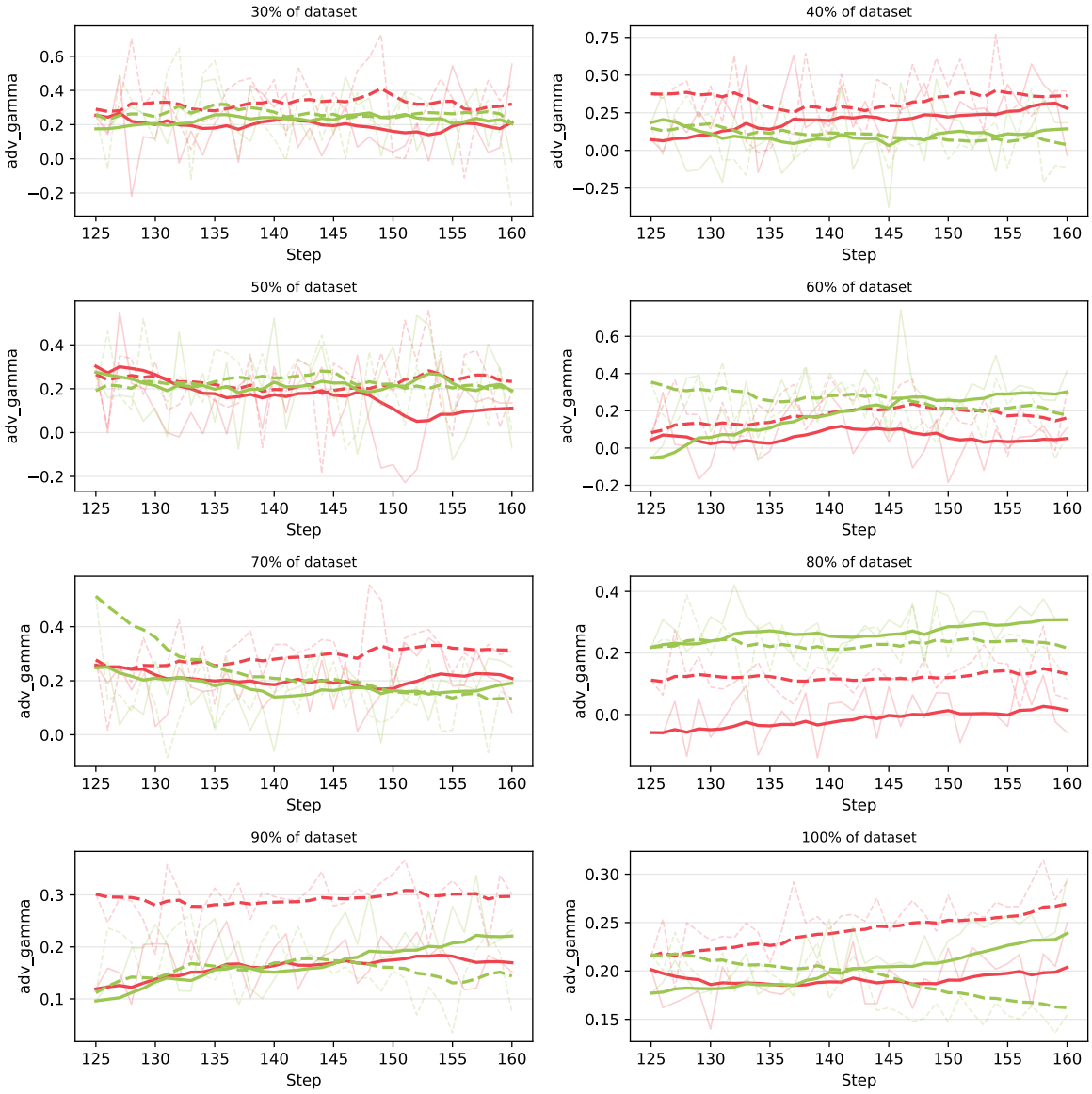

Figure 4: Elastic response of compression advantages over several sizes of datasets.

back" the model toward its pre-training behavior on non-watermarked data while pushing it away from the radioactive optimum.

Conclusion

Our result necessitates more conservative threat models for dataset watermarking. Existing analyses often do not consider adversaries who systematically exploit elasticity via targeted fine-tuning beyond naive model extraction or mild adaptation.[30] [31] Since generic fine-tuning suffices to erase radioactive signatures, then watermarking alone may be insufficient as a legal or technical mechanism for ownership verification in adversarial environments.

Additionally, radioactive watermarks can be interpreted as an extreme, highly concentrated form of *behavioral alignment* focused on a narrow distribution. Understanding how easily such concentrated alignment can be erased offers insight into the stability of more benign alignment signals, such as safety or preference tuning, which are also implemented via small datasets. If both are governed by the same elastic geometry, this would suggest that robust alignment requires either more extensive alignment data or architectural interventions that modify the optimization landscape.[32]

While our current experiments weaken but do not fully erase the radioactive advantage, they already show that elasticity-aware fine-tuning can substantially reduce detection power without catastrophic loss in generic performance. This suggests that future watermark schemes need to be evaluated not only against generic finetuning, but also against adversaries that actively exploit compression geometry.

Appendix

Full Proof of Theorem 1

We proceed in three steps.

Distribution of the loss score. By assumption, each per-example gradient $G(z)$ is distributed as $\mathcal{N}(0, \sigma^2 I_d)$. For a fixed unit vector v with $\|v\| = 1$, the scalar score $S = v^\top G$ is a linear functional of a multivariate Gaussian, hence $\mathbb{E}[S] = \mathbb{E}[v^\top G] = v^\top \mathbb{E}[G] = 0$, and $\mathrm{Var}(S) = \mathbb{E}[(v^\top G)^2] = \mathbb{E}[v^\top G G^\top v] = v^\top \mathbb{E}[GG^\top] v = v^\top (\sigma^2 I_d) v = \sigma^2$. Thus $S \sim \mathcal{N}(0, \sigma^2)$.

[30] B. G. A. Tekgul and N. Asokan, "On the Effectiveness of Dataset Watermarking in Adversarial Settings," *Proceedings of the 2022 ACM on Asia Conference on Computer and Communications Security (ASIACCS)*, pp. 1243–1245, 2022.

[31] W. Bouaziz, N. Usunier, and E.-M. El-Mhamdi, "Data Taggants: Dataset Ownership Verification via Harmless Targeted Data Poisoning," *International Conference on Learning Representations (ICLR)*, 2025.

[32] J. Ji, K. Wang, T. Qiu, et al., "Language Models Resist Alignment: Evidence From Data Compression," *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2025.

Define the standardized variable $X = S/\sigma$, so $X \sim \mathcal{N}(0,1)$. Let c_q be the unique value satisfying $\mathbb{P}(X \geq c_q) = q$. By definition of the standard normal cdf Φ, this means $1-\Phi(c_q) = q$, hence $c_q = \Phi^{-1}(1-q)$. The corresponding threshold in the original scale is $\tau_q = \sigma c_q$, so the event $\{S \geq \tau_q\}$ is equivalent to $\{X \geq c_q\}$.

Conditional expectation of the gradient under top-q selection. The pair (G, S) is jointly Gaussian, and the conditional mean of G given S is linear in S. We first compute $\mathrm{Cov}(G, S)$. Since $S = v^\top G$, we have $\mathrm{Cov}(G, S) = \mathbb{E}[GS] = \mathbb{E}[Gv^\top G] = \mathbb{E}[GG^\top]v = \sigma^2 I_d v = \sigma^2 v$. We already know $\mathrm{Var}(S) = \sigma^2$. For jointly Gaussian variables, the conditional mean satisfies $\mathbb{E}[G \mid S] = \mathrm{Cov}(G, S)\mathrm{Var}(S)^{-1}S$, so here $\mathbb{E}[G \mid S] = (\sigma^2 v)(1/\sigma^2)S = vS$.

To obtain $\mathbb{E}[G \mid S \geq \tau_q]$, we integrate this conditional mean over the truncated distribution of S: $\mathbb{E}[G \mid S \geq \tau_q] = \mathbb{E}[\mathbb{E}[G \mid S] \mid S \geq \tau_q] = \mathbb{E}[vS \mid S \geq \tau_q] = v\mathbb{E}[S \mid S \geq \tau_q]$. Using $S = \sigma X$ and the equivalence $\{S \geq \tau_q\} \Leftrightarrow \{X \geq c_q\}$, this becomes $\mathbb{E}[S \mid S \geq \tau_q] = \sigma\mathbb{E}[X \mid X \geq c_q]$.

The conditional density of X given $X \geq c_q$ is $f_{X|X\geq c_q}(x) = \phi(x)/(1-\Phi(c_q))$ for $x \geq c_q$, where $\phi(x)$ is the standard normal pdf. Therefore

$$\mathbb{E}[X \mid X \geq c_q] = (1/(1-\Phi(c_q))) \int_{c_q}^{\infty} x\phi(x)dx.$$

To evaluate the integral, we use the fact that $\phi'(x) = -x\phi(x)$, so $\int_{c_q}^{\infty} x\phi(x)dx = -\int_{c_q}^{\infty} \phi'(x)dx = \phi(c_q)$. Substituting this back gives $\mathbb{E}[X \mid X \geq c_q] = \phi(c_q)/(1-\Phi(c_q))$. By construction, $1-\Phi(c_q) = q$, hence $\mathbb{E}[X \mid X \geq c_q] = \phi(c_q)/q$, and consequently $\mathbb{E}[S \mid S \geq \tau_q] = \sigma\phi(c_q)/q$.

Finally, we obtain $\mathbb{E}[G \mid S \geq \tau_q] = v\mathbb{E}[S \mid S \geq \tau_q] = \sigma(\phi(c_q)/q)v$. Defining

$$\lambda(q) = \frac{\phi(\Phi^{-1}(1-q))}{q},$$

we can rewrite this as $\mathbb{E}[G_t] = \mathbb{E}[G \mid S \geq \tau_q] = \sigma\lambda(q)v$, which proves part (1) of the theorem.

Effect on the compression advantage. Under gradient descent on D_t, the infinitesimal parameter dynamics are $d\theta/d\tau = -\mathbb{E}[G_t] = -\sigma\lambda(q)v$. The chain rule gives $d\gamma_{D_w}(\theta(\tau))/d\tau = \nabla_\theta \gamma_{D_w}(\theta(\tau))^\top (d\theta/d\tau)$. At $\tau = 0$, we have $\theta(0) = \theta_w$ and by assumption $\nabla_\theta \gamma_{D_w}(\theta_w) = -\|g_w\|v$, so the derivative becomes $(-\|g_w\|v)^\top(-\sigma\lambda(q)v) = \|g_w\|\sigma\lambda(q)$.

The compression advantage is defined as $\Delta\gamma_{D_w}(\theta) = \gamma_{D_w}(\theta_0) - \gamma_{D_w}(\theta)$, where θ_0 is the base (unwatermarked) model. Differentiating with respect to τ yields $d\Delta\gamma_{D_w}(\theta(\tau))/d\tau = -d\gamma_{D_w}(\theta(\tau))/d\tau$, so at $\tau = 0$ we obtain

$$\frac{d}{d\tau}\Delta\gamma_{D_w}(\theta(\tau))\Big|_{\tau=0} = -\|g_w\|\sigma\lambda(q),$$

which is strictly negative whenever $\lambda(q) > 0$. This verifies part (3). The cross-elasticity coefficient is $E_{w,t} := \nabla_\theta \gamma_{D_w}(\theta_w)^\top \mathbb{E}[G_t] = (-\|g_w\| v)^\top (\sigma\lambda(q) v) = -\|g_w\| \sigma\lambda(q) < 0$, which completes part (2).

For the special case $q = 1/2$, we have $c_{1/2} = \Phi^{-1}(1/2) = 0$, hence $\phi(c_{1/2}) = \phi(0) = 1/\sqrt{2\pi}$ and $\lambda(1/2) = \phi(0)/(1/2) = \sqrt{2/\pi}$. Substituting this into the general expression above produces the explicit prediction $\frac{d}{d\tau} \Delta\gamma_{D_w}(\theta(\tau))\big|_{\tau=0} = -\|g_w\| \sigma \sqrt{2/\pi}$.