METRIC-NORMALIZED POSTERIOR LEAKAGE (MPL): ATTACKER-ALIGNED PRIVACY FOR JOINT CONSUMPTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Metric differential privacy (mDP) extends *local differential privacy (LDP)* by scaling noise to semantic distance, but many machine learning (ML) systems are consumed under joint observation, where model-agnostic, per-record guarantees can miss leakage from evidence aggregation. We introduce *metric-normalized posterior leakage (mPL)*, an attacker-aligned, distance-calibrated measure of posterior-odds shift induced by releases, and show that for single or independent releases, uniformly bounding mPL is equivalent to mDP. Under joint observation, however, satisfying mDP may still leave mPL high because learned aggregators compound evidence across correlated items. To make control practical, we formalize *probabilistically bounded mPL (PBmPL)*, which limits how often mPL may exceed a target budget, and we operationalize it via *Adaptive mPL (AmPL)*, a trust-and-verify framework that perturbs, audits with a learned attacker, and adapts parameters (with optional Bayesian remapping) to balance privacy and utility. In a word-embedding case study, a learned joint adversary reveals mPL violations under joint consumption despite per-record mDP perturbations, whereas AmPL substantially lowers the frequency of such violations with low utility loss, indicating PBmPL as a practical, auditable protection for joint-consumption settings.

1 INTRODUCTION

Local differential privacy (LDP) (Duchi et al., 2013) enforces a uniform indistinguishability requirement between any two inputs, regardless of how similar those inputs are. This metric-agnostic stance is often a poor fit for modern continuous domains and embedding spaces (Imola et al., 2022), where distances encode semantics: nearby points are naturally similar, while far-apart points may correspond to qualitatively different meanings. As a result, uniform protection can lead to an unfavorable utility–privacy trade-off, either injecting excessive noise or failing to adequately protect fine-grained neighborhoods.

Metric-aware privacy notions address this mismatch by incorporating geometry into the guarantee. In particular, *metric differential privacy (mDP)* (Chatzikokolakis et al., 2013) requires that the indistinguishability between secrets degrades smoothly with their metric distance, a formulation that is especially natural for locations and embeddings. In the location-privacy literature, *geo-indistinguishability* (Andrés et al., 2013) popularized practical mechanisms (e.g., planar Laplace noise) and motivated optimization-based obfuscation tailored to road networks, points of interest, and mobility priors. Subsequent work (Bordenabe et al., 2014; Liu & Qiu, 2025) studied optimal mechanism design under metric constraints, related utility to transport/Wasserstein-style costs, and developed composition and group-privacy analyses in metric spaces. Complementing these defenses, recent studies have analyzed inference risks under correlated releases and proposed context-aware perturbation methods for metric-aware guarantees (Qiu et al., 2022; Yadav et al., 2024).

Although mDP incorporates geometry, it is typically formulated as a bound on output-distribution ratios between pairs of inputs and is often analyzed on a per-release basis. In contrast, practical adversaries reason about *posterior beliefs* and routinely aggregate *correlated* observations, for example, multiple perturbed locations along a trajectory (Yadav et al., 2024) or multiple perturbed tokens within a sentence (Staab et al., 2024). These settings suggest that per-release guarantees alone may not faithfully reflect inferential risk under joint consumption. Accordingly, we shift the

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

evaluation target from isolated output ratios to the extent to which an attacker's beliefs *sharpen* after jointly observing multiple, potentially dependent releases, particularly in the presence of learned inference models that can effectively combine such evidence.

OUR CONTRIBUTIONS

(**1**) **From per-release ratio bounds to metric-normalized inferential leakage.** We rethink mDP for *joint-consumption* settings by introducing *metric-normalized posterior leakage (mPL)*, a *geometry-aware*, *attacker-aligned* criterion that quantifies the metric-calibrated shift in posterior odds between candidate secrets after observing releases (*Definition 2*). We define *bounded mPL* by requiring mPL to lie within a budget ϵ (*Definition 3*). Importantly, for a single release and independent compositions, we prove that uniformly bounding mPL is equivalent to ϵ-mDP (*Propositions 1–2*), showing that mPL *recovers* mDP in the regime where per-release analysis applies.

(**2**) **Joint-consumption leakage under dependence with learned evidence aggregation.** Under dependent joint consumption, the equivalence between per-record mDP and joint-observation mPL no longer holds. Although each marginal release channel may satisfy mDP, aggregating correlated releases can substantially sharpen an attacker's beliefs about a target secret, resulting in large mPL. We establish this gap with an exact two-secret construction. In realistic high-dimensional settings, however, the true joint prior is unknown, and exact inference under an unstructured model requires summing over an exponentially large space. We therefore fit a structured joint prior from auxiliary data and combine it with the known release likelihood through Bayes' rule to audit mPL. In a synthetic setting where the true joint distribution and exact posterior are available, we evaluate this approximation by comparing fitted-posterior and exact-posterior mPL violation ratios and privacy–utility outcomes.

(**3**) **Auditable tail-risk control and attacker-in-the-loop calibration.** To relax worst-case control under dependence, we introduce *probabilistically bounded mPL (PBmPL)* (Definition 4), which bounds, under a specified audit distribution, the probability that mPL exceeds a target budget. We accompany this definition with a sampling-based audit. Building on the audit, *Adaptive mPL (AmPL)* (i) applies level-wise perturbation, (ii) fits a structured joint prior from auxiliary data and combines it with the known release likelihood through Bayes' rule to audit mPL, and (iii) adapts the perturbation strengths to balance privacy and utility. AmPL can optionally apply *Bayesian remapping* (Chatzikokolakis et al., 2017) as post-processing to reduce perturbation-induced utility loss.

(**4**) **Case study: joint-consumption privacy for text embeddings.** We instantiate the framework on text embeddings, jointly protecting *personally identifiable information (PII)* and *potentially identifiable information (PoII)*. Across three text datasets, a learned-prior Bayesian attacker detects joint-observation mPL-budget exceedances even when each record is perturbed by a per-record mDP exponential mechanism. AmPL substantially reduces the measured pre-remapping exceedance rate across the evaluated dataset–budget settings while incurring modest utility loss.

2 METRIC-NORMALIZED POSTERIOR LEAKAGE

We first introduce the setting and mDP (§2.1) and define mPL and characterize its properties, including equivalence to mDP for single/independent releases (§2.2, §2.3). We then discuss joint consumption with correlated secrets, where per-record mDP may fail to control mPL (§2.4).

2.1 PRELIMINARIES

We study *local perturbation* mechanisms that randomize each record before release. Formally, a mechanism $\mathcal{M}$ is a randomized mapping $\mathcal{M} : \mathcal{X} \to \mathcal{Y}$, where $\mathcal{X}$ is the domain of secret records and $\mathcal{Y}$ is the domain of released (perturbed) records. The semantic similarity between secrets is captured by a distance function $d : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_{\geq 0}$; we write d_{x_i,x_j} for the distance between any $x_i, x_j \in \mathcal{X}$. We consider *joint consumption* of L secret records, represented as $\mathbf{x} = (x^1, \ldots, x^L) \in \mathcal{X}^L$. For each $\ell \in [L]$, let X^ℓ and Y^ℓ denote the random variables corresponding to the ℓ-th secret record and its released output, respectively. We write C for context information available to the attacker before release.

Definition 1 (mDP). *Let $(\mathcal{X}, d)$ be a metric secret space and let $\mathcal{M}$ be a perturbation mechanism with input space $\mathcal{X}$ and output space $\mathcal{Y}$. We say that a mechanism $\mathcal{M}$ satisfies* (ϵ, d)-mDP *if,* $\sup_{x_i \neq x_j} \sup_{y \in \mathcal{Y}} \ln \frac{\Pr[\mathcal{M}(x_i)=y]}{\Pr[\mathcal{M}(x_j)=y]} \leq \epsilon d_{x_i,x_j}$, *where ϵ denotes the* privacy budget.

Intuitively, mDP requires that small changes in input x induce only bounded changes in the law of $\mathcal{M}(x)$, yielding privacy calibrated to the metric d. A smaller ϵ implies a tighter bound, and hence stronger privacy, so that less can be inferred about x from observing $\mathcal{M}(x)$.

Following (Wang et al., 2017; Liu & Qiu, 2025; Imola et al., 2022), we consider a discrete perturbation space $\mathcal{Y} = \{y_1, \ldots, y_K\}$. To facilitate analysis, we represent $\mathcal{M}$ as a deterministic function $\tilde{\mathcal{M}}$ defined by $\mathcal{M}(x) \equiv \tilde{\mathcal{M}}(x, Z)$, where $Z \sim \mathrm{Uniform}(0, 1)$ is an auxiliary random variable that captures the randomness of $\mathcal{M}$. Specifically, we define cumulative sums

$$F_0(x) = 0, \quad F_k(x) = \textstyle\sum_{v=1}^{k} \Pr[\mathcal{M}(x) = y_v],\ k = 1, \ldots, K. \tag{1}$$

Then $\tilde{\mathcal{M}}$ is given by $\tilde{\mathcal{M}}(x, Z) = \sum_{k=1}^{K} y_k \, \mathbf{1}_{[F_{k-1}(x),\, F_k(x))}(Z)$, where $\mathbf{1}_{[a,b)}(Z)$ is the indicator function, equal to 1 if $Z \in [a, b)$ and 0 otherwise.

In the following, we use both notations $\mathcal{M}$ and $\tilde{\mathcal{M}}$: $\mathcal{M}$ for simplicity of exposition, and $\tilde{\mathcal{M}}$ in formal arguments (e.g., the proof of Proposition 2).

Threat model. We adopt standard assumptions (Liu & Qiu, 2025): the server is *honest-but-curious*, follows the protocol, and attempts inference. The attacker is prior-informed, knows the mechanism $\mathcal{M}$ (and its parameters), observes public context C, has auxiliary data from the same population to fit a joint prior, and can passively aggregate one or more correlated noisy releases to infer x_ℓ from a candidate set $\mathcal{X}_\ell$. Given at least a single perturbed record y and the mechanism $\mathcal{M}$, the server can infer the posterior distribution of X using Bayes' rule (Yu et al., 2017),

$$\Pr(X = x \mid Y = y, C) = \frac{\Pr(Y = y \mid X = x, C) \Pr(X = x \mid C)}{\sum_{x' \in \mathcal{X}} \Pr(Y = y \mid X = x', C) \Pr(X = x' \mid C)}. \tag{2}$$

This adversarial model reflects realistic deployments. The server is operated by an organization with incentives to collect and analyze user data, and thus cannot be fully trusted from a privacy perspective. In contrast, users are assumed to be honest and to faithfully apply the prescribed perturbation protocol before transmitting their data.

Joint observation. In this paper, we consider attackers who observe *multiple* perturbed releases. Given a user's secret sequence $\mathbf{x} = (x^{(1)}, \ldots, x^{(L)})$, a randomized mechanism $\mathcal{M}$ is applied *independently* to each component, producing $\mathbf{y} = (y^{(1)}, \ldots, y^{(L)})$ with $y^{(\ell)} = \mathcal{M}\left(x^{(\ell)}\right)$ for $\ell \in [L]$. The adversary observes the joint output $\mathbf{y}$ and aims to infer the original secrets $\mathbf{x}$. Importantly, while the perturbations are applied independently across ℓ, the secrets $\{x^{(\ell)}\}_{\ell=1}^{L}$ may be statistically dependent. The attacker estimates each secret marginal conditioned on the public context and the full released sequence:

$$\Pr(X_\ell = x \mid C, \mathbf{Y} = \mathbf{y}), \qquad x \in \mathcal{X}. \tag{3}$$

2.2 mPL and Its Post-Processing Property

To quantify the information contributed by the joint release, we compare posterior and pre-release odds for the same public context C.

Definition 2 (mPL). *For distinct candidates x_i, x_j at sensitive position ℓ, the metric-normalized posterior leakage given $\mathbf{Y} = \mathbf{y}$ is*

$$\mathrm{mPL}_{\mathcal{M}}(x_i, x_j, \mathbf{y}; C) = \frac{1}{d_{x_i, x_j}} \left| \log \frac{\Pr(X_\ell = x_i \mid C, \mathbf{Y} = \mathbf{y})}{\Pr(X_\ell = x_j \mid C, \mathbf{Y} = \mathbf{y})} - \log \frac{\Pr(X_\ell = x_i \mid C)}{\Pr(X_\ell = x_j \mid C)} \right|. \tag{4}$$

Figure 1 illustrates the shift from the prior $\Pr(X_\ell = x_i \mid C)$ to the posterior $\Pr(X_\ell = x_i \mid C, \mathbf{Y} = \mathbf{y})$ after observing the joint release. A smaller belief shift indicates that the perturbation mechanism $\mathcal{M}$ reveals less information, thereby offering stronger privacy protection.

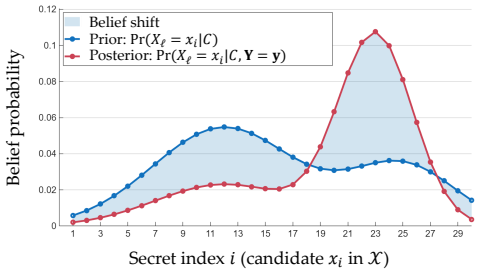

Figure 1: Attacker belief update: prior vs. posterior distribution.

Notably, both mPL (**Definition 2**) and the posterior-based formulation of Kifer & Machanavajjhala (2012) rely on Bayesian belief updating; the main distinction lies in the privacy criterion. mPL measures the metric-normalized change in posterior odds between candidate values of a target secret X_ℓ under a joint release. In contrast, Pufferfish bounds posterior-odds changes for specified secret pairs uniformly over an explicit class of data-generating distributions, without requiring metric normalization. Thus, mPL provides a metric-aware, position-level criterion tailored to local joint-consumption settings.

Definition 3 (ϵ-Bounded mPL)**.** *Assume positive candidate priors and strictly positive release likelihoods on the admissible output space. A randomized mechanism $\mathcal{M}$ satisfies ϵ-bounded mPL if, for every admissible public context C, target position, and distinct candidate pair,*

$$\sup_{x_i \neq x_j} \sup_{\mathbf{y} \in \mathcal{Y}^L} \mathrm{mPL}_{\mathcal{M}}(x_i, x_j, \mathbf{y}; C) \leq \epsilon. \tag{5}$$

2.3 Properties Based on Individual or Independent Observations

In this part, we first show that mPL exactly recovers mDP for a single observation, and then extend this equivalence to joint observations when the secret–noise pairs are conditionally independent.

Proposition 1 (Single-observation equivalence of mPL and mDP)**.** *Fix an admissible public context C. Assume a positive candidate prior and strictly positive channel likelihoods. For a pair (x_i, x_j) and observation y, define*

$$\mathrm{mPL}_{\mathcal{M}}((x_i, x_j), y; C) = \frac{1}{d_{x_i, x_j}} \left| \ln \frac{\Pr(X = x_i \mid C, \mathcal{M}(X) = y)}{\Pr(X = x_j \mid C, \mathcal{M}(X) = y)} - \ln \frac{\Pr(X = x_i \mid C)}{\Pr(X = x_j \mid C)} \right|. \tag{6}$$

For any $\epsilon \geq 0$, the conditional channel satisfies (ϵ, d)-mDP if and only if

$$\sup_{x_i \neq x_j} \sup_{y \in \mathcal{Y}} \mathrm{mPL}_{\mathcal{M}}((x_i, x_j), y; C) \leq \epsilon.$$

A proof appears in Appendix C.1.

We next consider secrets that are conditionally independent given C. Under this assumption, we establish a connection between individual and joint posterior leakage in *Proposition 2*:

Proposition 2 (Independent-observation equivalence of mPL and mDP)**.** *If the secret–noise pairs (X_ℓ, Z_ℓ) are independent across positions conditional on C, then joint-observation mPL at each position equals its single-observation mPL. Hence a uniform joint mPL bound of ϵ is equivalent to the corresponding conditional per-record (ϵ, d)-mDP bound. A proof appears in Appendix C.2.*

The proposition shows that without inter-record dependencies, individual-level mDP bounds suffice to ensure privacy under joint observation. However, this assumption rarely holds in practice.

2.4 Threat Models based on Joint and Correlated Observations

In this part, we relax the independence assumption and introduce more realistic threat models where the records $X_1, \ldots, X_L$ are dependent.

(1) Explicit joint-probability attacker (a toy example). We consider an attacker that models the *joint* distribution of two secrets and performs *Bayesian inference* over two perturbed outputs. Let $X_1, X_2 \in \mathcal{X} = \{x_1, x_2\}$ with a correlated prior: $\Pr(X_1 = x_1, X_2 = x_1) = \Pr(X_1 = x_2, X_2 = x_2) = 0.01$ and $\Pr(X_1 = x_1, X_2 = x_2) = \Pr(X_1 = x_2, X_2 = x_1) = 0.49$, and set $\epsilon = 1.0$. For an *exponential mechanism (EM)* perturbation $\mathcal{M}_{\mathrm{EM}}$ with two outputs $\{y_1, y_2\}$, suppose

$$\Pr(\mathcal{M}_{\mathrm{EM}}(X_i) = y_1 \mid X_i = x_1) = 0.72, \quad \Pr(\mathcal{M}_{\mathrm{EM}}(X_i) = y_2 \mid X_i = x_1) = 0.28,$$

$$\Pr(\mathcal{M}_{\mathrm{EM}}(X_i) = y_1 \mid X_i = x_2) = 0.28, \quad \Pr(\mathcal{M}_{\mathrm{EM}}(X_i) = y_2 \mid X_i = x_2) = 0.72.$$

A direct calculation shows that for each $y_k \in \{y_1, y_2\}$, $\mathrm{mPL}_{\mathcal{M}_{\mathrm{EM}}}(x_1, x_2, y_k) = 0.944 < \epsilon$, so observing each perturbed record *individually* does not violate the mPL bound (therefore also achieving mDP according to Proposition 1). In contrast, when the two outputs are consumed *jointly*, we obtain $\mathrm{mPL}_{\mathcal{M}_{\mathrm{EM}}}\big(x_1, x_2, \{\mathcal{M}_{\mathrm{EM}}(x_1), \mathcal{M}_{\mathrm{EM}}(x_2)\} = \{y_1, y_2\}\big) = 1.846 > \epsilon$, demonstrating that joint consumption under a correlated prior can trigger posterior-leakage violations even when all single-observation checks pass. Full details appear in Appendix B.2.

(2) Learned joint-distribution attacker. Explicit Bayesian joint inference can, in principle, reveal joint leakage. For an unstructured joint distribution, direct posterior normalization requires summing over $|\mathcal{X}|^L$ configurations, which is exponential in the number of sensitive positions L. Moreover, the joint prior is typically unknown and must be estimated from data (Bernardo & Smith, 1994). We therefore use a learned mixture of structured chain distributions. Prior and posterior marginals are computed by applying standard forward–backward inference to each chain component (Kschischang et al., 2001; Sutton & McCallum, 2012) and combining the component marginals using the corresponding mixture weights before or after conditioning on the release (Appendix E.2).

Attacker overview. The attacker models sensitive tokens conditional on the public context C by combining neural and count-based components. Records are partitioned in proportions of $50/15/15/10/10$ for *training*, *validation*, a *reserved holdout*, *mechanism selection*, and *testing*. The fitted attacker remains fixed during mechanism selection and final held-out evaluation. Appendix E.2 provides the model, training, and marginal- inference details.

Prior and posterior approximation. Let $\widehat{q}_N$ and $\widehat{q}_C$ denote the fitted neural and count-based joint distributions. With mixture weight w selected by validation joint likelihood, the attacker defines the fitted prior

$$\widehat{q}(\mathbf{X} \mid C) = (1-w)\widehat{q}_N(\mathbf{X} \mid C) + w\widehat{q}_C(\mathbf{X} \mid C). \tag{7}$$

After observing the joint release $\mathbf{y}$, the attacker computes

$$\widehat{p}_{\boldsymbol{\alpha}}(\mathbf{X} \mid C, \mathbf{y}) \propto \widehat{q}(\mathbf{X} \mid C) \prod_{j=1}^{L} \mathcal{M}_{g(X_j)}\big(y_j \mid X_j; \alpha_{g(X_j)}\big). \tag{8}$$

Figure 2 summarizes the learned-prior Bayesian attacker. The attacker fits the joint prior $\widehat{q}(\mathbf{X} \mid C)$ using auxiliary data consisting of clean secret record set $\mathcal{D}_{\text{aux}}$ and their associated public contexts; these auxiliary samples are disjoint from the target secret records $\mathcal{D}_{\text{tar}}$. For a target record $\mathbf{x}$ from $\mathcal{D}_{\text{tar}}$, the mechanism $\mathcal{M}$ generates the observed joint release $\mathbf{y}$ from the unknown secret $\mathbf{x}$. Given the target context C, the observed release, and the known release likelihood, the attacker performs a Bayesian update to obtain the posterior marginal $\widehat{p}_{\boldsymbol{\alpha}}(X_\ell \mid C, \mathbf{y})$ at each target position ℓ. The fitted prior marginal $\widehat{q}(X_\ell \mid C)$ and the resulting posterior marginal provide the pairwise prior and posterior log odds used in the mPL audit.

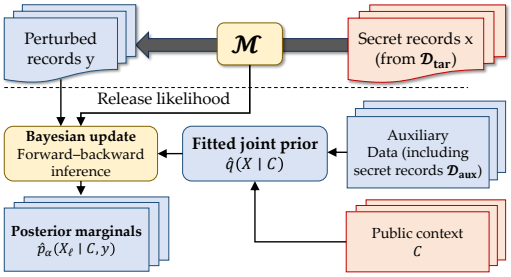

Figure 2: Fitted prior and release-conditioned posterior over candidate secrets at target position ℓ. The joint release redistributes the attacker's belief across candidates.

The release likelihoods enter each structured component as local evidence, and forward–backward inference computes the corresponding marginals and evidence probabilities. The validation-selected prior weight w remains fixed, while conditioning induces release-dependent posterior component weights. The neural component is trained first by joint likelihood and then by a weighted conditional-likelihood objective. Appendix E.2 provides the parameterization, training settings, inference procedure, and numerical conventions used in the audit.

Discussion: Posterior-approximation gap. The posterior in Eq. (8) is an exact Bayesian update under the fitted joint prior, but it may differ from the posterior under the unknown true distribution. Without assumptions on model misspecification, this gap is difficult to characterize formally or quantify on real-world data. **Proposition 4** in Section 3 provides a conditional guarantee: If the fitted prior and posterior approach their true counterparts in KL divergence, and candidate probabilities and distances are bounded away from zero, then the expected mPL estimation error also approaches zero at the square-root rate of the KL approximation error. Because these conditions and constants are generally unknown in practice, we complement this result with a synthetic evaluation for which the exact posterior is computable. As detailed in Table 2 in our case study (Section 4), the estimated and exact evaluations produce consistent conclusions, with only small differences in violation ratios and utility losses.

Discussion: Per-user accounting. Per-user budgeting is effective when reliable user identifiers enable per-user accounting; we instead target settings without such identifiers (e.g., many text/embedding datasets) and adopt a user-agnostic formulation that controls joint leakage over arbitrarily correlated secrets (detailed discussion can be found in Appendix B.3).

3 Data Perturbation Framework

As discussed in Section 2.4, exact closed-form calibration of mechanism parameters for mPL is generally intractable, since evaluating mPL requires posteriors induced by high-dimensional, correlation-aware likelihoods. To operationalize mPL, we adopt a *trust–and–verify* framework with an attacker in the loop, called *Adaptive mPL (AmPL)*. AmPL starts from a principled per-record

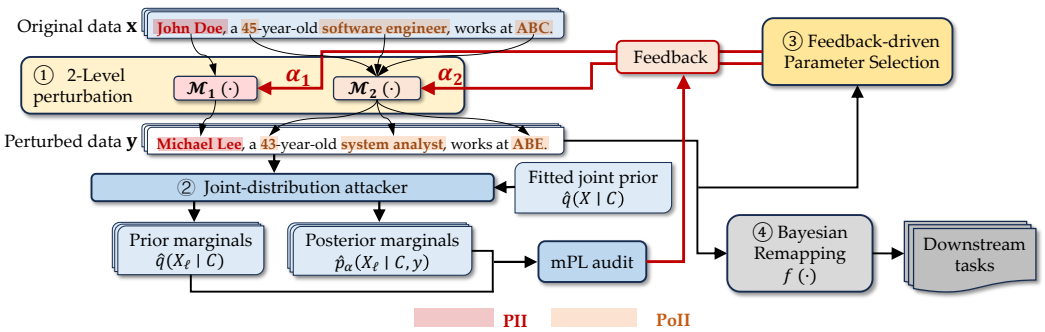

Figure 3: Illustration of the AmPL framework (example: protecting PII and PoII word embeddings).

perturbation (motivated by the independent case), fits a learned joint attacker before parameter selection, and then *verifies* candidate perturbation strengths by auditing the resulting mPL estimates. AmPL uses this audit feedback to select the mechanism parameters. Finally, AmPL supports *level-wise* protection by stratifying secrets into multiple sensitivity tiers.

Figure 3 illustrates the AmPL framework via a text-embedding example with two sensitivity tiers: *personally identifiable information (PII)* and *potentially identifiable information (PoII)*. Here, PII includes attributes that directly identify or authenticate an individual (e.g., full name, email address, phone number, or precise home address). PoII refers to attributes that may not uniquely identify a person in isolation but can materially reduce the anonymity set or reveal sensitive traits when combined with other data (e.g., employer, city of residence, demographic descriptors, or fine-grained preferences). AmPL here perturbs the two tiers at different strengths, uses the learned joint attacker to audit mPL, and selects perturbation strengths from that feedback. Bayesian remapping is applied to improve utility.

Next, we introduce the details of Steps ①–④ within the AmPL framework.

Step ①: Level-wise data perturbation. Let $N \in \mathbb{N}$ denote the number of sensitivity tiers. We first partition $\mathcal{X}$ into disjoint subsets $\{\mathcal{X}^{(1)}, \ldots, \mathcal{X}^{(N)}\}$ and define a level-assignment function $g : \mathcal{X} \to \{1, \ldots, N\}$ that maps each secret $x \in \mathcal{X}$ to its sensitivity level. For each level $\ell \in \{1, \ldots, N\}$, specify a mechanism $\mathcal{M}_\ell(\cdot; \alpha_\ell)$ with privacy/perturbation parameter α_ℓ. Given x, the released output is $y \sim \mathcal{M}_{g(x)}\big(x; \alpha_{g(x)}\big)$, so that protection strength matches the sensitivity of x. The collection $\{\alpha_\ell\}_{\ell=1}^N$ can be tuned (i.e., via feedback control in Step ③) to meet target leakage–utility trade-offs.

Two-level example for word-embedding privacy (PII vs. PoII): In this case, we let $N = 2$ with $\mathcal{X}^{(1)}$ denoting direct identifiers (PII) and $\mathcal{X}^{(2)}$ denoting quasi-identifiers (PoII). We assign a stronger perturbation to PII and a milder one to PoII (e.g., mechanisms $\mathcal{M}_1, \mathcal{M}_2$ with $\epsilon_1 < \epsilon_2$), reflecting their different disclosure risks. A detailed design and evaluation of this two-level word-embedding perturbation appear in Section 4 (Case Study) and Appendix D.

Step ②: Learned Adversary. The learned attacker described in Section 2.4 supplies a prior $\widehat{q}(X_\ell \mid C)$ and a release-updated posterior $\widehat{p}(X_\ell \mid C, \mathbf{y})$ for each sensitive position. We compare their log odds across candidate pairs to audit posterior leakage. Requiring the mPL bound to hold for every candidate pair and every possible release yields a worst-case criterion that may require excessive perturbation. We therefore introduce the probabilistic relaxation in Definition 4, which bounds, under a specified audit distribution, the probability that mPL exceeds the target budget.

Definition 4 (Probabilistic bounded mPL)**.** *Given a perturbation mechanism $\mathcal{M}_\ell$ ($\ell = 1, ..., L$), we define the violation probability $p_{\mathcal{X}_\ell^2}$ as the probability that the posterior leakage exceeds the privacy budget ϵ for a randomly sampled pair of records and output: $p_{\mathcal{X}_\ell^2} = \Pr\left[\mathrm{mPL}_{\mathcal{M}_\ell}(X_i, X_j, Y) > \epsilon\right]$, where X_i and X_j are drawn from the secret record domain $\mathcal{X}_\ell$. We say $\mathcal{M}_\ell$ can achieve (δ, ϵ_ℓ)-PBmPL if $p_{\mathcal{X}_\ell^2} \leq \delta$.*

Directly computing $p_{\mathcal{X}_\ell^2}$ is computationally prohibitive, as it requires evaluating all $|\mathcal{X}_\ell|^2$ record pairs. In practice, where $|\mathcal{X}_\ell|$ may range from tens of thousands to over one hundred thousand records (e.g., 5,448 PIIs and 5,492 PoIIs in AG-News dataset (Zhang et al., 2015)), exhaustive evaluation is intractable. To address this, we estimate $p_{\mathcal{X}_\ell^2}$ via *random sampling*. Specifically, we uniformly

sample a subset $\mathcal{S}_\ell \subseteq \mathcal{X}_\ell^2 \times \mathcal{Y}$ consisting of S_ℓ triplets (x_i, x_j, y), and define the empirical estimate:

$$\hat{p}_{\mathcal{S}_\ell} = \frac{1}{S_\ell} \sum_{(x_i,x_j,y)\in\mathcal{S}_\ell} \mathbf{1}\left(\mathrm{mPL}_{\mathcal{M}_\ell}(x_i, x_j, y) > \epsilon\right), \tag{9}$$

where $\mathbf{1}(\cdot)$ denotes the indicator function. Because $\mathcal{S}_\ell$ is sampled uniformly, $\hat{p}_{\mathcal{S}_\ell}$ is an *unbiased estimator* of $p_{\mathcal{X}_\ell^2}$, i.e., $\mathbb{E}[\hat{p}_{\mathcal{S}_\ell}] = p_{\mathcal{X}_\ell^2}$. We further establish the following concentration guarantee:

Proposition 3 (Concentration guarantee for PBmPL sampling)**.** *If the empirical violation rate satisfies $\hat{p}_{\mathcal{S}_\ell} = \xi\delta$ for some constant $\xi < 1$, then $\Pr\left[p_{\mathcal{X}_\ell^2} \le \delta\right] \ge 1 - 2\exp(-2S_\ell(1-\xi)^2\delta^2)$. The detailed proof can be found in **Appendix C.3**.*

Proposition 3 shows that as the sample size S_ℓ increases, the bound $2\exp(-2S_\ell(1-\xi)^2\delta^2)$ rapidly approaches zero, ensuring that $\Pr[p_{\mathcal{X}_\ell^2} \le \delta]$ approaches one.

Proposition 4 (Stability of the mPL audit)**.** *For a fixed sensitive position, let $p_0(\cdot \mid C)$ and $p_1(\cdot \mid C, \mathbf{Y})$ be the true pre-release prior and posterior marginals, and let $\widehat{p}_0, \widehat{p}_1$ be their fitted counterparts. Assume a finite candidate space, that all four distributions assign each candidate probability at least $\gamma > 0$, and that distinct candidates have distance at least $d_{\min} > 0$. Let n denote the number of auxiliary training records used to fit the joint prior from which $\widehat{p}_0$ and $\widehat{p}_1$ are derived. If $\mathbb{E}_{C,\mathbf{Y}}[\mathrm{KL}(p_0\|\widehat{p}_0) + \mathrm{KL}(p_1\|\widehat{p}_1)] \le C_0 n^{-\alpha}$, then, for every fixed candidate pair (x_i, x_j),*

$$\mathbb{E}_{C,\mathbf{Y}}\left[\left|\mathrm{mPL}(x_i, x_j; C, \mathbf{Y}) - \widehat{\mathrm{mPL}}(x_i, x_j; C, \mathbf{Y})\right|\right] \le \frac{2\sqrt{C_0}}{\gamma d_{\min}}\, n^{-\alpha/2}.$$

Proposition 4 accounts for errors in both sides of the prior-to-posterior odds comparison. Under its stated KL condition, the expected mPL estimation error decreases at rate $n^{-\alpha/2}$; the proof appears in Appendix C.4.

Step ③: Feedback-Driven Perturbation Adjustment. To balance privacy protection and utility preservation, we adopt an adaptive optimization strategy that iteratively updates the scaling factors α_1 and α_2 based on adversarial feedback. This adaptation is guided by minimizing a composite loss function $\mathcal{L}(\boldsymbol{\alpha})$, which jointly captures privacy leakage and utility degradation:

$$\mathcal{L}(\boldsymbol{\alpha}) = \lambda_1 \cdot \mathcal{L}_{\mathrm{privacy}}(\boldsymbol{\alpha}) + \lambda_2 \cdot \mathcal{L}_{\mathrm{utility}}(\boldsymbol{\alpha}), \tag{10}$$

where $\lambda_1, \lambda_2 > 0$ are trade-off coefficients that balance the two objectives. The privacy loss term $\mathcal{L}_{\mathrm{privacy}}(\boldsymbol{\alpha})$ is given by the empirical violation rate $\hat{p}_{\mathcal{S}}$, which estimates the probability that posterior leakage exceeds the privacy budget ϵ over a sampled set of input-output pairs. The utility loss term $\mathcal{L}_{\mathrm{utility}}(\boldsymbol{\alpha})$ represents the expected semantic distortion caused by the perturbation: $\mathcal{L}_{\mathrm{utility}}(\boldsymbol{\alpha}) = \sum_{\ell=1}^{L} \sum_{x\in\mathcal{X}_1} \sum_{y\in\mathcal{Y}} \pi_x\, c_{x,y}\, \Pr\left(\mathcal{M}_\ell(x;\alpha_\ell) = y\right)$, where π_x denotes the prior probability of x, and $c_{x,y}$ quantifies the utility loss incurred by reporting y when the true input is x.

Step ④: Bayesian Remapping. While perturbation mechanisms protect privacy by injecting noise into sensitive data, the resulting outputs may not be optimal for downstream tasks due to semantic distortion. To mitigate this utility loss, we employ *Bayesian remapping*, a post-processing step that refines perturbed outputs based on their inferred posterior distributions. Given a perturbed record y, Bayesian remapping selects an alternative output that minimizes the expected utility loss under the posterior, formally defined as:

$$f(y) = \arg\min_{y'\in\mathcal{Y}} \sum_{\ell=1}^{L} \sum_{x\in\mathcal{X}_\ell} \underbrace{\Pr\left[X = x \mid \mathcal{M}_\ell(x;\alpha_\ell) = y\right]}_{\text{posterior of } x \text{ given perturbed record } y} c_{x,y'}. \tag{11}$$

Notably, Chatzikokolakis et al. (2017) showed that Bayesian remapping preserves the original mDP guarantee for individual perturbed observations. We further show in Proposition 5 that bounded joint mPL is preserved under post-processing. However, this invariance applies to bounded mPL and does not directly imply preservation of PBmPL; we therefore empirically re-audit the remapped mechanism, with details provided in **Appendix B.1**.

4 Case Study: PII/PoII Embedding Protection

We evaluate AmPL on text embeddings because embeddings are widely used in deployed systems, and text exhibits layered sensitivities (PII vs. PoII) that naturally align with level-wise perturbation;

Table 1: Posterior leakage violation ratio (%).

Method	$\varepsilon = 2.40$	$\varepsilon = 2.50$	$\varepsilon = 2.60$
		AG News	
EM (mDP)	0.000343 ± 0.000337	0.000402 ± 0.000355	0.000494 ± 0.000386
AmPL	0.000076 ± 0.000048	0.000115 ± 0.000080	0.000121 ± 0.000083
AmPL-P	0.000000 ± 0.000000	0.000000 ± 0.000000	0.000000 ± 0.000000
		IMDB Reviews	
EM (mDP)	0.000333 ± 0.000409	0.000347 ± 0.000541	0.000329 ± 0.000745
AmPL	0.000066 ± 0.000122	0.000052 ± 0.000045	0.000061 ± 0.000053
AmPL-P	0.000000 ± 0.000000	0.000000 ± 0.000000	0.000000 ± 0.000000
		Amazon Reviews	
EM (mDP)	0.000345 ± 0.000320	0.000356 ± 0.000311	0.000328 ± 0.000199
AmPL	0.000082 ± 0.000100	0.000141 ± 0.000229	0.000172 ± 0.000168
AmPL-P	0.000000 ± 0.000000	0.000000 ± 0.000000	0.000000 ± 0.000000

Table 2: Exact and estimated privacy–utility outcomes on the synthetic world.

ε	Measure	EM		AmPL	
		Estimated	Exact	Estimated	Exact
2.2	VR (%)	0.95282 ± 0.03005	1.22944 ± 0.00291	0.00005 ± 0.00004	0.00009 ± 0.00006
	Utility, pre	0.09023 ± 0.00035	0.09069 ± 0.00017	0.35371 ± 0.02077	0.35637 ± 0.02000
	Utility, post	0.08032 ± 0.00042	0.08092 ± 0.00024	0.33020 ± 0.01783	0.33287 ± 0.01693
3	VR (%)	0.35518 ± 0.00633	0.40966 ± 0.00141	0.00011 ± 0.00005	0.00016 ± 0.00007
	Utility, pre	0.01578 ± 0.00011	0.01516 ± 0.00008	0.29814 ± 0.01745	0.30269 ± 0.01686
	Utility, post	0.01353 ± 0.00011	0.01291 ± 0.00008	0.27736 ± 0.01623	0.28217 ± 0.01561
3.5	VR (%)	0.21011 ± 0.00311	0.22256 ± 0.00028	0.00008 ± 0.00005	0.00012 ± 0.00006
	Utility, pre	0.00529 ± 0.00012	0.00490 ± 0.00006	0.26421 ± 0.00728	0.26954 ± 0.00720
	Utility, post	0.00451 ± 0.00013	0.00412 ± 0.00007	0.24469 ± 0.00713	0.25036 ± 0.00708
4	VR (%)	0.16223 ± 0.00272	0.16666 ± 0.00017	0.00014 ± 0.00010	0.00019 ± 0.00013
	Utility, pre	0.00192 ± 0.00006	0.00170 ± 0.00003	0.22965 ± 0.01827	0.23471 ± 0.01836
	Utility, post	0.00166 ± 0.00006	0.00144 ± 0.00003	0.21137 ± 0.01761	0.21683 ± 0.01771

text embeddings also support standard threat models (joint posterior inference) and utility evaluation through candidate replacement costs.

Although our experiments perturb data in embedding space, mPL/AmPL are modality- and representation-agnostic, requiring only a metric over secret records, a utility loss, and a learned attacker (see Appendix E.12 and Appendix B.4); additional case-study details are provided in Appendix D.

We use the pre-trained GloVe embeddings (Pennington et al., 2014) and evaluate on three benchmark text datasets: (1) *AG-News* (Zhang et al., 2015), a four-class news classification corpus with 120,000 training and 7,600 test samples (World, Sports, Business, and Sci/Tech); (2) *IMDB* (Maas et al., 2011), a binary sentiment dataset of 50,000 reviews evenly split between positive and negative labels; and (3) the *Amazon Reviews* corpus (Zhang et al., 2015), a large-scale collection spanning multiple domains, containing 34,686,770 reviews from 6,643,669 users over 2,441,053 products.

We use the Exponential Mechanism (EM) for perturbation, which samples directly from a finite candidate set with a distance-aligned utility score, typically yielding high utility for discrete outputs (Feyisetan et al., 2020). In particular, we apply two *adjusted EM* perturbation mechanisms, $\mathcal{M}_{\mathrm{EM}}(\cdot;\alpha_1\epsilon)$ and $\mathcal{M}_{\mathrm{EM}}(\cdot;\alpha_2\epsilon)$, to protect PII and PoII, controlled by scaling factors α_1 and α_2 (with $\alpha_1, \alpha_2 \in [0,1]$ and $\alpha_1 < \alpha_2$). Formally, for $\ell \in \{1,2\}$,

$$\Pr\left[\mathcal{M}_{\mathrm{EM}}(x;\alpha_\ell\epsilon) = y\right] = \frac{\exp\left(-\frac{1}{2}\alpha_\ell\epsilon \cdot d_{x,y}\right)}{\sum_{y' \in \mathcal{Y}} \exp\left(-\frac{1}{2}\alpha_\ell\epsilon \cdot d_{x,y'}\right)}, \tag{12}$$

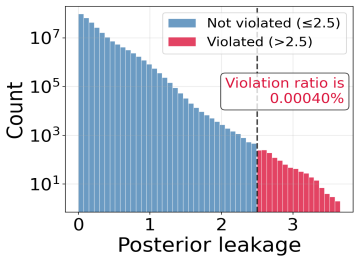

Figure 4: Example of mPL distributions.

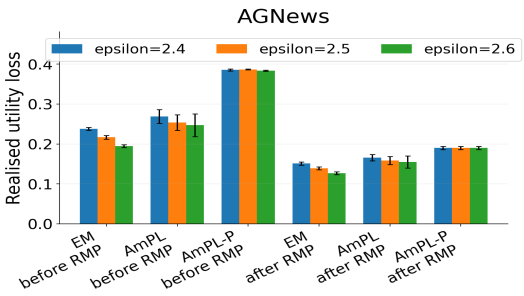

Figure 5: Utility loss.

where $\mathcal{M}_1(\cdot)$ (with smaller α_1) introduces stronger noise for PII, and $\mathcal{M}_2(\cdot)$ (with larger α_2) applies milder noise for PoII. As a baseline, we use the standard EM in Eq. (12) with no level differentiation, i.e., $\alpha_1 = \alpha_2 = 1$. For ablations, we compare our full method (AmPL) against (i) *AmPL-P* (privacy-preserving), which sets $\lambda_2 = 0$ in the objective of Eq. (10), thereby optimizing only privacy loss; and (ii) *AmPL-I*, which applies identity-salience weighting to PII only (no PoII weighting).

Main results. Table 1 shows that AmPL lowers the pre-remapping violation ratio relative to EM in all nine dataset–budget settings under the learned joint attacker. The choice of $\epsilon \in \{2.4, 2.5, 2.6\}$ is discussed in Appendix E.9. On AG News at $\epsilon = 2.50$, for example, the ratio falls from 0.000402% for EM to 0.000115% for AmPL. Figure 4 helps explain what the aggregate ratio measures: although most EM leakage values remain below the budget, a small right tail crosses it. The reduction under AmPL shows that feedback from joint posterior inference can suppress these tail events, which per-record mDP alone does not control under correlated observations. Additional distributions are reported in Appendix E.3. The corresponding mPL violation ratios after Bayesian remapping are discussed in Appendix B.1

Because exact posteriors are unavailable for real-world data, we construct a correlated synthetic setting in which the true joint distribution is known. Each record contains eight sensitive tokens and six public-context tokens, whose dependencies are governed by a latent topic. We compare the audit based on a joint prior fitted from auxiliary data with an exact-posterior evaluation computed from the generator parameters. Both use the same exponential release mechanism and are evaluated at $\epsilon \in \{2.2, 3.0, 3.5, 4.0\}$. Full generator, inference, and data-split details appear in Appendix E.4. Table 2 shows that evaluation under the fitted posterior preserves the conclusions obtained under the exact posterior. Across all four budgets, both evaluations rank AmPL below EM in violation ratio and find that remapping reduces utility loss. For AmPL, the fitted- and exact-posterior violation ratios differ by less than 5×10^{-5} percentage points, while the corresponding utility losses differ by less than 0.006 in the reported cost units. At $\epsilon = 2.2$, near the peak of the baseline leakage curve, the exact-posterior violation ratio decreases from 1.229435% for EM to 0.000090% for AmPL. These results support the fitted posterior as a practical basis for leakage auditing and perturbation-strength selection in this setting.

Finally, Figure 5 compares utility loss on AG News before and after remapping. At $\epsilon = 2.50$, remapping lowers the mean loss to 0.1392 for EM, 0.1586 for AmPL, and 0.1901 for AmPL-P. Results for the other datasets appear in Appendix E.7; the attacker-training-size analysis appears in Appendix E.10.

Further experimental results, including the *detailed synthetic evaluation of posterior approximation, effect of PoII protection on PII leakage, empirical PBmPL thresholds, embedding-level utility-loss and downstream-classification comparisons, privacy–utility trade-offs, wider ϵ ranges, sensitivity to attacker training-data size, robustness to attacker-side embedding-channel mismatch, and a non-text tabular-data case study*, are reported in Appendix E.4–E.12.

5 Conclusions

We formalized metric-normalized posterior leakage and its relaxation (PBmPL), and proposed AmPL, an adaptive attacker-aligned perturbation mechanism. Across three text datasets under a learned joint attacker, AmPL reduces the measured pre-remapping violation ratio with low utility loss. Future work will extend AmPL beyond text, strengthen PBmPL composition, and evaluate broader adversaries.

AI USE STATEMENT

We used a large language model (LLM) assistant to aid wording, improve clarity, and polish exposition. The LLM was not used to generate novel results, code, or citations, and no outputs were accepted without human review. The authors verified the accuracy of all assisted text and take full responsibility for the final content.

ETHICS STATEMENT

This work is intended to help system designers identify and reduce inferential privacy leakage from correlated releases. However, the learned-prior Bayesian attacker developed for auditing is dual-use: the same inference procedure could be applied to infer sensitive information from deployed systems. It should therefore be used only for authorized evaluation and defensive mechanism calibration. We report aggregate attack and leakage measurements and do not attempt to identify, contact, or reconstruct information about particular individuals.

The framework assumes access to auxiliary data containing clean sensitive records and associated public context. In a real deployment, collecting and storing such data may itself create privacy risks. Practitioners should obtain an appropriate legal and ethical basis for its use, minimize collection and retention, enforce access controls, and keep auxiliary and target records separate. The empirical PBmPL audit is relative to its specified audit distribution and fitted joint prior; a low measured violation rate should not be interpreted as a universal privacy guarantee or evidence of regulatory compliance. Model misspecification, distribution shift, or a stronger adversary may produce greater leakage than the audit detects.

Our experiments use existing benchmark datasets under their stated terms and licenses; we collect no new participant data and conduct no interventions with human subjects. Automated PII/PoII identification based on named-entity recognition and embedding similarity is an imperfect proxy that may behave differently across languages, demographic groups, and cultural contexts. Consequently, the proposed sensitivity tiers should be reviewed by domain experts and affected stakeholders before deployment.

REPRODUCIBILITY STATEMENT

The paper provides the definitions and proofs of mPL and PBmPL, the perturbation and audit procedures, and the distance and utility measures used in evaluation. Sections 2–4 and Appendices C–E describe the attacker model, Bayesian inference, sensitivity-tier construction, numerical safeguards, experimental settings, and compute resources; Appendix F lists the data and software sources and their terms of use. For the text experiments, records are partitioned into training, validation, reserved holdout, mechanism-selection, and test sets in proportions of 50/15/15/10/10. Attacker checkpoints and the prior-mixture weight are selected on validation data, and the fitted attacker is held fixed during mechanism selection and final evaluation. Reported violation ratios are measured before remapping, while utility is evaluated both before and after remapping. The synthetic study separately specifies its data generator and splits; exact posterior quantities are used only for evaluation, not for selecting the deployable mechanism.

REFERENCES

Miguel E Andrés, Nicolás E Bordenabe, Konstantinos Chatzikokolakis, and Catuscia Palamidessi. Geo-indistinguishability: Differential privacy for location-based systems. In *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, pp. 901–914, 2013.

José M. Bernardo and Adrian F. M. Smith. *Bayesian Theory*. Wiley Series in Probability and Statistics. John Wiley & Sons, Chichester, UK, 1994. ISBN 9780471494645. doi: 10.1002/9780470316870.

N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi. Optimal geo-indistinguishable mechanisms for location privacy. In *Proc. of ACM CCS*, pp. 251–262, 2014.

Konstantinos Chatzikokolakis, Miguel E Andrés, Nicolás Emilio Bordenabe, and Catuscia Palamidessi. Broadening the scope of differential privacy using metrics. In *international symposium on privacy enhancing technologies symposium*, pp. 82–102. Springer, 2013.

Kostas Chatzikokolakis, Ehab Elsalamouny, and Catuscia Palamidessi. Efficient utility improvement for location privacy. *Proceedings on Privacy Enhancing Technologies*, 2017, 10 2017. doi: 10.1515/popets-2017-0051.

John C. Duchi, Michael I. Jordan, and Martin J. Wainwright. Local privacy and statistical minimax rates. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 429–438. IEEE, 2013. doi: 10.1109/FOCS.2013.53. Full version: arXiv:1302.3203.

O. Feyisetan, B. Balle, T. Drake, and T. Diethe. Privacy- and utility-preserving textual analysis via calibrated multivariate perturbations. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, pp. 178–186, New York, NY, USA, 2020. Association for Computing Machinery.

Fadi Hassan, David Sánchez, and Josep Domingo-Ferrer. Utility-preserving privacy protection of textual documents via word embeddings. *IEEE Transactions on Knowledge and Data Engineering*, 35(1):1058–1071, 2023. doi: 10.1109/TKDE.2021.3076632.

Jacob Imola, Shiva Kasiviswanathan, Stephen White, Abhinav Aggarwal, and Nathanael Teissier. Balancing utility and scalability in metric differential privacy. In *Proc. of UAI 2022*, 2022.

Daniel Kifer and Ashwin Machanavajjhala. A rigorous and customizable framework for privacy. In *Proc. of the 31st ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, PODS '12, pp. 77–88, New York, NY, USA, 2012. Association for Computing Machinery. ISBN 9781450312486. doi: 10.1145/2213556.2213571. URL https://doi.org/10.1145/2213556.2213571.

Frank R Kschischang, Brendan J Frey, and H-A Loeliger. Factor graphs and the sum-product algorithm. *IEEE Transactions on information theory*, 47(2):498–519, 2001.

P. Langley. Crafting papers on machine learning. In Pat Langley (ed.), *Proceedings of the 17th International Conference on Machine Learning (ICML 2000)*, pp. 1207–1216, Stanford, CA, 2000. Morgan Kaufmann.

Ruiyao Liu and Chenxi Qiu. Panda: Rethinking metric differential privacy optimization at scale with anchor-based approximation. In *Proceedings of The 32nd ACM Conference on Computer and Communications Security (CCS)*, 2025.

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

Sebastian Meiser. Approximate and probabilistic differential privacy definitions. Cryptology ePrint Archive, Paper 2018/277, 2018. URL https://eprint.iacr.org/2018/277.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, 2014.

Chenxi Qiu, Li Yan, Anna Squicciarini, Juanjuan Zhao, Chengzhong Xu, and Primal Pappachan. Trafficadaptor: an adaptive obfuscation strategy for vehicle location privacy against traffic flow aware attacks. In *Proceedings of the 30th International Conference on Advances in Geographic Information Systems*, pp. 1–10, 2022.

Robin Staab, Mark Vero, Mislav Balunovic, and Martin Vechev. Beyond memorization: Violating privacy via inference with large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=kmn0BhQk7p.

Daniel W. Stroock. *Probability Theory: An Analytic View*. Cambridge University Press, 2nd edition, 2010. ISBN 0521132509.

Charles Sutton and Andrew McCallum. An introduction to conditional random fields. *Foundations and Trends® in Machine Learning*, 4(4):267–373, 2012.

L. Wang, D. Yang, X. Han, T. Wang, D. Zhang, and X. Ma. Location privacy-preserving task allocation for mobile crowdsensing with differential geo-obfuscation. In *Proc. of ACM WWW*, pp. 627–636, 2017.

William Wolberg, Olvi Mangasarian, Nick Street, and W. Street. Breast Cancer Wisconsin (Diagnostic). UCI Machine Learning Repository, 1993. URL https://doi.org/10.24432/C5DW2B.

Sourabh Yadav, Chenyang Yu, Xinpeng Xie, Yan Huang, and Chenxi Qiu. Protecting vehicle location privacy with contextually-driven synthetic location generation. SIGSPATIAL '24, pp. 29–41, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400711077. doi: 10.1145/3678717.3691211. URL https://doi.org/10.1145/3678717.3691211.

Lei Yu, Ling Liu, and Calton Pu. Dynamic differential location privacy with personalized error bounds. In *NDSS*, volume 17, pp. 1–15, 2017.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28, 2015.

APPENDIX

Appendix Overview. Appendix A consolidates key mathematical notation. Appendix B provides additional discussions. Appendix C contains omitted proofs. Appendix D provides additional case-study details. Appendix E reports extended experiments. Appendix F summarizes licenses and terms of use for existing assets. Appendix G provides the impact statement.

A MATH NOTATIONS

Table 3: Notation summary. Symbols are grouped by role (spaces/variables, mechanisms/metrics, leakage/audit quantities, and optimization).

Symbol	Description
Spaces and random variables	
$\mathcal{X}, \mathcal{Y}$	Secret (embedding) space and perturbed/release space.
$x_i \in \mathcal{X}, y_k \in \mathcal{Y}$	A candidate secret embedding and a candidate perturbed embedding.
X, X_ℓ	Random variables for a secret embedding and the ℓ-th secret in a sequence.
C	Public information available before the release.
$\widehat{q}(\mathbf{X} \mid C)$	Fitted joint prior over sensitive tokens conditional on C.
$\widehat{p}(\mathbf{X} \mid C, \mathbf{Y})$	Posterior induced by the fitted prior and the release mechanism.
d_{x_i,x_j}	Metric distance on $\mathcal{X}$ between x_i and x_j.
c_{x_i,y_k}	Utility loss when releasing y_k for true input x_i.
Mechanisms and parameters	
$\mathcal{M}$	Perturbation mechanism mapping $\mathcal{X} \to \mathcal{Y}$.
$\mathcal{M}(\cdot, \alpha_\ell)$ $(\ell = 1, ..., N)$	Level-wise mechanisms for N sensitivity tiers (e.g., PII vs. PoII).
$\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_N)$	Vector of per-level perturbation strengths.
$f_t(y)$	Tier-indexed remapping applied to a released token.
Leakage and audit quantities	
$\mathrm{mPL}_{\mathcal{M}}(x_i, x_j, \mathbf{y}; C)$	Metric-normalized change from prior odds given C to posterior odds given $(C, \mathbf{y})$.
ϵ	Target mPL budget (smaller is more private).
$p_{\mathcal{A}}$	Violation probability under a specified audit distribution $\mathcal{A}$.
$\widehat{v}_{\mathrm{test}}$	Held-out pooled violation ratio.
δ	Tolerance on violation frequency for PBmPL.
Optimization and adaptation (AmPL)	
$\mathcal{L}(\boldsymbol{\alpha})$	Composite objective balancing privacy and utility.
$\mathcal{L}_{\text{privacy}}(\boldsymbol{\alpha})$	Pooled pre-remapping violation ratio on the selection split.
$\mathcal{L}_{\text{utility}}(\boldsymbol{\alpha})$	Expected utility distortion averaged over sensitive occurrences on the selection split.
λ_1, λ_2	Weights trading off privacy vs. utility in $\mathcal{L}(\cdot)$.

Table 4: Attack results under different privacy budgets. Each entry reports the violation ratio (%) before and after Bayesian remapping (mean $\pm$ std).

Dataset	$\epsilon = 2.40$	$\epsilon = 2.50$	$\epsilon = 2.60$
AG News	$(7.62 \pm 2.47) \times 10^{-5} \rightarrow (3.16 \pm 3.71) \times 10^{-5}$	$(1.15 \pm 0.41) \times 10^{-4} \rightarrow (6.62 \pm 4.67) \times 10^{-5}$	$(1.21 \pm 0.42) \times 10^{-4} \rightarrow (7.42 \pm 4.39) \times 10^{-5}$
IMDB	$(6.61 \pm 6.22) \times 10^{-5} \rightarrow (1.93 \pm 2.10) \times 10^{-6}$	$(5.20 \pm 2.28) \times 10^{-5} \rightarrow (2.39 \pm 4.23) \times 10^{-6}$	$(6.08 \pm 2.70) \times 10^{-5} \rightarrow (3.92 \pm 5.43) \times 10^{-6}$
Amazon Review	$(8.16 \pm 5.11) \times 10^{-5} \rightarrow (3.13 \pm 3.01) \times 10^{-6}$	$(1.41 \pm 1.17) \times 10^{-4} \rightarrow (1.55 \pm 3.14) \times 10^{-5}$	$(1.72 \pm 0.85) \times 10^{-4} \rightarrow (2.33 \pm 3.27) \times 10^{-5}$

B Discussions

B.1 Post-Processing Property of mPL

Of theoretical interest, we establish post-processing invariance for bounded mPL in Proposition 5.

Proposition 5 (Post-processing invariance for bounded mPL)**.** *Let $\mathcal{M} : \mathcal{X} \to \mathcal{Y}$ be a randomized mechanism that satisfies the ϵ-bounded mPL constraint. For any (possibly randomized) function $f : \mathcal{Y} \to \mathcal{Z}$, define the post-processed mechanism $(f \circ \mathcal{M})(x) \triangleq f(\mathcal{M}(x))$, $\forall x \in \mathcal{X}$, where $\mathcal{Z} = \mathrm{Range}(f \circ \mathcal{M})$. Then $f \circ \mathcal{M}$ also satisfies the bounded joint mPL constraint:* $\sup_{x_i \neq x_j} \sup_{\mathbf{z} \in \mathcal{Z}^L} \mathrm{mPL}_{f\circ\mathcal{M}}(x_i, x_j, \mathbf{z}) \leq \epsilon$.

Proof. Let $f^{-1}(\mathbf{z}) = \{\mathbf{y} : f(\mathbf{y}) = \mathbf{z}\}$ denote the preimage of z. Fix any pair $x_i, x_j \in \mathcal{X}$ and any $\mathbf{y} \in \mathcal{Y}^L$, we have

$$
\begin{aligned}
& \mathrm{mPL}_{f\circ\mathcal{M}}(x_i, x_j, \mathbf{z}) && (13)\\
= \; & \frac{1}{d_{x_i,x_j}} \left| \ln \left(\frac{\Pr(X_\ell = x_i \,|\, \{f \circ \mathcal{M}(X_1), \ldots, f \circ \mathcal{M}(X_L)\} = \mathbf{z})}{\Pr(X_\ell = x_j \,|\, \{f \circ \mathcal{M}(X_1), \ldots, f \circ \mathcal{M}(X_L)\} = \mathbf{z})} \Big/ \frac{\Pr(X_\ell = x_i)}{\Pr(X_\ell = x_j)} \right) \right| && (14)\\
= \; & \frac{1}{d_{x_i,x_j}} \left| \ln \left(\frac{\Pr(X_\ell = x_i, \{\mathcal{M}(X_1), \ldots, \mathcal{M}(X_L)\} = f^{-1}(\mathbf{z}))}{\Pr(X_\ell = x_j, \{\mathcal{M}(X_1), \ldots, \mathcal{M}(X_L)\} = f^{-1}(\mathbf{z}))} \Big/ \frac{\Pr(X_\ell = x_i)}{\Pr(X_\ell = x_j)} \right) \right| && (15)\\
\leq \; & \epsilon. && (16)
\end{aligned}
$$

□

Intuitively, Proposition 5 states that if the joint mPL is bounded by ϵ for every possible output sequence $\mathbf{y}$, then applying a data-independent transformation $f(\mathbf{y})$ cannot increase this worst-case privacy loss. This is because post-processing only maps or aggregates outcomes in the output space; it does not introduce additional access to the sensitive input. Therefore, the distinction between $X_\ell = x_i$ and $X_\ell = x_j$ cannot be amplified under the bounded mPL guarantee.

Notably, however, this post-processing invariance result applies to bounded mPL and does not directly extend to PBmPL. The reason is that bounded mPL is a pointwise worst-case guarantee, whereas PBmPL is a probabilistic relaxation that allows a small fraction of outputs to violate the mPL bound. A post-processing step may change how outputs are grouped, remapped, or reweighted, and therefore may change the probability of the violation event (Meiser, 2018). Thus, PBmPL should not be assumed to be automatically preserved under Bayesian remapping as a direct consequence of Proposition 5.

Table 4 reports a separate post-remapping diagnostic for AmPL under the learned joint attacker. In the nine reported dataset–budget settings, the observed violation ratio decreases after remapping.

These results suggest that, although PBmPL preservation should not be claimed as a general post-processing property, Bayesian remapping empirically improves the audited PBmPL violation ratio in our experiments. Accordingly, privacy claims after remapping are supported by a fresh empirical audit of the remapped mechanism rather than inherited from Proposition 5.

B.2 Inference Models based on Explicit Joint Probability

We construct a scenario, in which (1) two secret records X_1 and X_2 are not independently distributed, (2) observing each perturbed record ($\mathcal{M}_{\mathrm{EM}}(X_1)$ or $\mathcal{M}_{\mathrm{EM}}(X_2)$) *individually* doesn't violate the

posterior leakage bound, yet (3) observing $\mathcal{M}_{\mathrm{EM}}(X_1)$ and $\mathcal{M}_{\mathrm{EM}}(X_2)$ *jointly* causes a posterior leakage bound violation for X_1.

Suppose that X_1 and X_2 each take values in $\mathcal{X} = \{x_1, x_2\}$ with the following joint distribution:

$$\Pr(X_1 = x_1,\, X_2 = x_1) = 0.01, \tag{17}$$
$$\Pr(X_1 = x_1,\, X_2 = x_2) = 0.49, \tag{18}$$
$$\Pr(X_1 = x_2,\, X_2 = x_1) = 0.49, \tag{19}$$
$$\Pr(X_1 = x_2,\, X_2 = x_2) = 0.01. \tag{20}$$

Then each X_i has marginal distribution: $\Pr(X_i = x_1) = 0.5$ and $\Pr(X_i = x_2) = 0.5$. Therefore,

$$\Pr(X_1 = x_i,\, X_2 = x_j) \neq \Pr(X_1 = x_i)\Pr(X_2 = x_j), \tag{21}$$

$\forall x_i, x_j \in \mathcal{X}$ indicating X_1 and X_2 are not independent.

We let $\mathcal{M}_{\mathrm{EM}}(X_i) \in \{y_1, y_2\}$. The perturbation probabilities are given by

$$\Pr\left(\mathcal{M}_{\mathrm{EM}}(X_i) = y_1 \mid X_i = x_1\right) = 0.72, \tag{22}$$
$$\Pr\left(\mathcal{M}_{\mathrm{EM}}(X_i) = y_2 \mid X_i = x_1\right) = 0.28, \tag{23}$$
$$\Pr\left(\mathcal{M}_{\mathrm{EM}}(X_i) = y_1 \mid X_i = x_2\right) = 0.28, \tag{24}$$
$$\Pr\left(\mathcal{M}_{\mathrm{EM}}(X_i) = y_2 \mid X_i = x_2\right) = 0.72. \tag{25}$$

Finally, we set $d(x_1, x_2) = 1$ and the privacy budget $\epsilon = 1$.

(1) The posterior by observing each individual perturbed record: The *posterior odds* given the observation $\mathcal{M}_{\mathrm{EM}}(X_i) = y_1$ is

$$\left|\ln\left(\frac{\Pr(X_i = x_1 \mid \mathcal{M}_{\mathrm{EM}}(X_i) = y_1)}{\Pr(X_i = x_2 \mid \mathcal{M}_{\mathrm{EM}}(X_i) = y_1)} \Big/ \frac{\Pr(X_i = x_1)}{\Pr(X_i = x_2)}\right)\right| \tag{26}$$
$$= \left|\ln\left(\frac{\Pr(\mathcal{M}_{\mathrm{EM}}(X_i) = y_1 \mid X_i = x_1)}{\Pr(\mathcal{M}_{\mathrm{EM}}(X_i) = y_1 \mid X_i = x_2)}\right)\right| \tag{27}$$
$$= \left|\ln\left(\frac{0.72}{0.28}\right)\right| \tag{28}$$
$$= 0.9444 \tag{29}$$
$$< \epsilon. \tag{30}$$

Similarly, the *posterior odds* given the observation $\mathcal{M}_{\mathrm{EM}}(X_i) = y_2$ is

$$\left|\ln\left(\frac{\Pr(X_i = x_1 \mid \mathcal{M}_{\mathrm{EM}}(X_i) = y_2)}{\Pr(X_i = x_2 \mid \mathcal{M}_{\mathrm{EM}}(X_i) = y_2)} \Big/ \frac{\Pr(X_i = x_1)}{\Pr(X_i = x_2)}\right)\right| \tag{31}$$
$$= \left|\ln\left(\frac{\Pr(\mathcal{M}_{\mathrm{EM}}(X_i) = y_2 \mid X_i = x_1)}{\Pr(\mathcal{M}_{\mathrm{EM}}(X_i) = y_2 \mid X_i = x_2)}\right)\right| \tag{32}$$
$$= \left|\ln\left(\frac{0.28}{0.72}\right)\right| \tag{33}$$
$$= 0.9444 \tag{34}$$
$$< \epsilon. \tag{35}$$

indicating that $\left|\ln\left(\frac{\Pr(X_i=x_1|\mathcal{M}_{\mathrm{EM}}(X_i)=y_j)}{\Pr(X_i=x_2|\mathcal{M}_{\mathrm{EM}}(X_i)=y_j)} \Big/ \frac{\Pr(X_i=x_1)}{\Pr(X_i=x_2)}\right)\right| \leq 1$ for each X_i and y_k.

(2) Posterior leakage given joint observation: The *posterior odds* given the joint observation $\mathcal{M}_{\mathrm{EM}}(X_1) = y_1, \mathcal{M}_{\mathrm{EM}}(X_2) = y_2$ is

$$\left|\ln\left(\frac{\Pr(X_1 = x_1 \mid \mathcal{M}_{\mathrm{EM}}(X_1) = y_1,\, \mathcal{M}_{\mathrm{EM}}(X_2) = y_2)}{\Pr(X_1 = x_2 \mid \mathcal{M}_{\mathrm{EM}}(X_1) = y_1,\, \mathcal{M}_{\mathrm{EM}}(X_2) = y_2)} \Big/ \frac{\Pr(X_1 = x_1)}{\Pr(X_1 = x_2)}\right)\right| \tag{36}$$

$$= \left|\ln\left(\frac{\Pr(X_1 = x_1 \mid \mathcal{M}_{\mathrm{EM}}(X_1) = y_1,\, \mathcal{M}_{\mathrm{EM}}(X_2) = y_2)}{\Pr(X_1 = x_2 \mid \mathcal{M}_{\mathrm{EM}}(X_1) = y_1,\, \mathcal{M}_{\mathrm{EM}}(X_2) = y_2)} \Big/ 1\right)\right| \tag{37}$$

$$= \left|\ln\left(\frac{\Pr(X_1 = x_1, \mathcal{M}_{\mathrm{EM}}(X_1) = y_1,\, \mathcal{M}_{\mathrm{EM}}(X_2) = y_2)}{\Pr(X_1 = x_2, \mathcal{M}_{\mathrm{EM}}(X_1) = y_1,\, \mathcal{M}_{\mathrm{EM}}(X_2) = y_2)}\right)\right| \tag{38}$$

$$= \left|\ln\left(\frac{\sum_{x\in\{x_1,x_2\}} \Pr(_{\mathcal{M}_{\mathrm{EM}}(X_1)=y_1,\, \mathcal{M}_{\mathrm{EM}}(X_2)=y_2)}^{X_1=x_1, X_2=x,})}{\sum_{x\in\{x_1,x_2\}} \Pr(_{\mathcal{M}_{\mathrm{EM}}(X_1)=y_1,\, \mathcal{M}_{\mathrm{EM}}(X_2)=y_2)}^{X_1=x_2, X_2=x,})}\right)\right| \tag{39}$$

$$= \left|\ln\left(\begin{array}{c} \displaystyle\sum_{x\in\{x_1,x_2\}} \Pr(X_1 = x_1, X_2 = x) \\ \Pr(\mathcal{M}_{\mathrm{EM}}(X_1) = y_1,\, \mathcal{M}_{\mathrm{EM}}(X_2) = y_2 | X_1 = x_1, X_2 = x) \\ \hline \displaystyle\sum_{x\in\{x_1,x_2\}} \Pr(X_1 = x_2, X_2 = x) \\ \Pr(\mathcal{M}_{\mathrm{EM}}(X_1) = y_1,\, \mathcal{M}_{\mathrm{EM}}(X_2) = y_2 | X_1 = x_2, X_2 = x) \end{array}\right)\right| \tag{40}$$

$$= \left|\ln\left(\begin{array}{c} \displaystyle\sum_{x\in\{x_1,x_2\}} \Pr(X_1 = x_1, X_2 = x) \\ \Pr(\mathcal{M}_{\mathrm{EM}}(X_1) = y_1 | X_1 = x_1) \Pr(\mathcal{M}_{\mathrm{EM}}(X_2) = y_2 | X_2 = x) \\ \hline \displaystyle\sum_{x\in\{x_1,x_2\}} \Pr(X_1 = x_2, X_2 = x) \\ \Pr(\mathcal{M}_{\mathrm{EM}}(X_1) = y_1 | X_1 = x_2) \Pr(\mathcal{M}_{\mathrm{EM}}(X_2) = y_2 | X_2 = x) \end{array}\right)\right| \tag{41}$$

$$= \left|\ln\left(\frac{0.01 \times 0.72 \times 0.28 + 0.49 \times 0.72 \times 0.72}{0.49 \times 0.28 \times 0.28 + 0.01 \times 0.28 \times 0.72}\right)\right| \tag{42}$$

$$= 1.8456 \tag{43}$$

$$> \epsilon. \tag{44}$$

B.3 Per-User Accounting As a Complementary Mitigation

We note that when records can be cleanly grouped by user and the mechanism is explicitly designed with per-user accounting, a per-user privacy budget is a natural and effective way to mitigate composition across correlated records. In the classical DP setting, this corresponds to treating each user as the "unit of protection," ensuring that all contributions from the same user share a fixed budget.

However, *user grouping is not always known or reliable in the kinds of applications we target.* For example, in many text embedding-based systems, the mechanism does not have a trusted user identifier for each record. Posts may come from multiple accounts controlled by the same person, or a single account may refer to several different individuals or secrets. Determining that a set of words, embeddings, or snippets describe the same person or the same underlying secret is itself part of the adversarial inference task, for example, linking posts across accounts, or linking mentions of the same individual across different documents. In such settings, a per-user budget implicitly assumes that this partition into users is known and enforced by the defender, whereas our threat model explicitly allows the adversary to aggregate any correlated releases they can link. This issue is also reflected in our case study. The public datasets we use do not contain user identifiers or reliable group labels that could serve as a ground truth for "per-user" segmentation. Constructing a per-user baseline would therefore require introducing additional, task-specific grouping heuristics (e.g., clustering by text similarity), which are orthogonal to our core threat model. Since our goal is to study joint leakage over arbitrary correlated secrets, without assuming that the defender knows how records should be grouped, we deliberately adopt a user-agnostic formulation of mPL and AmPL.

On the other hand, we see per-user budgeting as a complementary mitigation, not as an alternative to our framework. In applications where reliable user identifiers and grouping assumptions are available, our mechanisms and audits can be combined with per-user accounting: the system can enforce a per-user privacy budget while our framework still evaluates whether correlated secrets within or across those groups violate the intended metric privacy guarantees.

B.4 Perturbation Space: Embeddings vs. Text.

Our current implementation instantiates AmPL by perturbing word embeddings, because embeddings are a common interface in modern NLP systems (e.g., for search, retrieval, and recommendation). However, the *framework* itself is agnostic to whether perturbations are applied in embedding space or directly on text.

Formally, let $\mathcal{X}$ denote the space of original text (words or sentences) and $\mathcal{Y}$ the space of perturbed outputs (which may be text or embeddings). Any defense mechanism that specifies a randomized map $\mathcal{M} : \mathcal{X} \to \mathsf{Distr}(\mathcal{Y}), \quad x \mapsto \mathcal{M}(\cdot \mid x)$ induces a perturbation method and a corresponding posterior $\Pr[X \mid Y]$. Our mPL definition and learned-adversary attack are defined purely in terms of this posterior, and therefore apply unchanged to *any* such stochastic channel.

In particular, defenses that act directly on text, such as token deletion, insertion of noise characters, or synonym substitution, still define a stochastic mapping from original text x to perturbed text y. An attacker observing y can then process it through the same embedding model (or any other feature extractor) and train a predictor exactly as in our experiments. From the perspective of mPL and the learned adversary, the only requirement is that (X, Y) are jointly distributed via some randomized mechanism; the choice of operating in embedding space or text space is an implementation detail of the defense, not a limitation of the framework.

C Omitted Proofs

C.1 Proof of Proposition 1 (Single-Observation Equivalence of mPL and mDP)

Proposition 1 (Single-observation equivalence of mPL and mDP)**.** *Fix an admissible public context C. Suppose every candidate has positive prior probability and the channel likelihood is strictly positive for every candidate and admissible output. Define*

$$\mathrm{mPL}_{\mathcal{M}}((x_i, x_j), y; C) = \frac{1}{d_{x_i,x_j}} \left| \ln \frac{\Pr(X = x_i \mid C, \mathcal{M}(X) = y)}{\Pr(X = x_j \mid C, \mathcal{M}(X) = y)} - \ln \frac{\Pr(X = x_i \mid C)}{\Pr(X = x_j \mid C)} \right|. \quad (45)$$

For any $\epsilon \geq 0$, the conditional channel satisfies (ϵ, d)-mDP if and only if

$$\sup_{x_i \neq x_j} \sup_{y \in \mathcal{Y}} \mathrm{mPL}_{\mathcal{M}}((x_i, x_j), y; C) \leq \epsilon.$$

Proof. Bayes' rule gives, for every candidate pair and output,

$$\frac{\Pr(X = x_i \mid C, \mathcal{M}(X) = y)}{\Pr(X = x_j \mid C, \mathcal{M}(X) = y)} \Big/ \frac{\Pr(X = x_i \mid C)}{\Pr(X = x_j \mid C)} = \frac{\Pr(\mathcal{M}(X) = y \mid X = x_i, C)}{\Pr(\mathcal{M}(X) = y \mid X = x_j, C)}.$$

Thus mPL is the absolute log channel-likelihood ratio divided by d_{x_i,x_j}. Bounding it by ϵ for all pairs and outputs is equivalent to the two ordered likelihood-ratio inequalities of (ϵ, d)-mDP. □

C.2 Proof of Proposition 2 (Independent-Observation Equivalence of mPL and mDP)

Proposition 2 (Independent-observation equivalence of mPL and mDP)**.** *Conditional on a fixed C, if the secret–noise pairs (X_ℓ, Z_ℓ) are independent across positions, then joint-observation mPL at each position equals its single-observation mPL. Consequently, the uniform joint mPL bound is equivalent to the conditional per-record mDP bound.*

Proof. Work under the conditional distribution given C. Independent (X_ℓ, Z_ℓ) pairs have independent perturbed outputs $\widetilde{\mathcal{M}}(X_\ell, Z_\ell)$ because measurable functions preserve independence (Stroock, 2010). In the following calculation, all probabilities are conditional on the fixed public context C; we suppress C from the notation.

$$\Pr\left[X_\ell = x \mid \left\{\tilde{\mathcal{M}}(X_1, Z_1), \ldots, \tilde{\mathcal{M}}(X_L, Z_L)\right\} = \mathbf{y}\right] \tag{46}$$

$$= \frac{\Pr\left[X_\ell = x, \left\{\tilde{\mathcal{M}}(X_1, Z_1), \ldots, \tilde{\mathcal{M}}(X_L, Z_L)\right\} = \mathbf{y}\right]}{\Pr\left[\left\{\tilde{\mathcal{M}}(X_1, Z_1), \ldots, \tilde{\mathcal{M}}(X_L, Z_L)\right\} = \mathbf{y}\right]} \tag{47}$$

$$= \frac{\prod_{t=1, t\neq\ell}^{L} \Pr\left[\tilde{\mathcal{M}}(X_t, Z_t) = y_t\right] \Pr\left[X_\ell = x, \tilde{\mathcal{M}}(X_\ell, Z_\ell) = y_\ell\right]}{\prod_{t=1}^{L} \Pr\left[\tilde{\mathcal{M}}(X_t, Z_t) = y_t\right]} \tag{48}$$

$$= \Pr\left[X_\ell = x \,\middle|\, \tilde{\mathcal{M}}(X_\ell, Z_\ell) = y_\ell\right] \tag{49}$$

Thus joint and single-observation mPL coincide at the target position, and Proposition 1 gives the stated mDP equivalence. □

C.3 Proof of Proposition 3 (Concentration Guarantees)

Proposition 3. *[Concentration guarantee for PBmPL sampling] If the empirical violation rate satisfies $\hat{p}_{\mathcal{S}_\ell} = \xi\delta$ for some constant $\xi < 1$, then $\Pr\left[p_{\mathcal{X}^2_\ell} \leq \delta\right] \geq 1 - 2\exp(-2S_\ell(1-\xi)^2\delta^2)$.*

Proof. Suppose the empirical violation rate satisfies $\hat{p}_{\mathcal{S}} = \xi\delta$ for some constant $\xi < 1$. Our goal is to show that, with high probability, the true violation probability $p_{\mathcal{X}^2}$ is at most δ.

First, by Hoeffding's inequality, for any $t > 0$:

$$\Pr\left[|\hat{p}_{\mathcal{S}} - p_{\mathcal{X}^2}| > t\right] \leq 2e^{-2S_\ell t^2}. \tag{50}$$

Let $t = (1-\xi)\delta$. Then:

$$\Pr\left[|\hat{p}_{\mathcal{S}} - p_{\mathcal{X}^2}| > (1-\xi)\delta\right] \leq 2e^{-2S_\ell(1-\xi)^2\delta^2}. \tag{51}$$

This implies:

$$\Pr\left[|\hat{p}_{\mathcal{S}} - p_{\mathcal{X}^2}| \leq (1-\xi)\delta\right] \geq 1 - 2e^{-2S_\ell(1-\xi)^2\delta^2}. \tag{52}$$

Now, under the event that $|\hat{p}_{\mathcal{S}} - p_{\mathcal{X}^2}| \leq (1-\xi)\delta$, we have:

$$p_{\mathcal{X}^2} \leq \hat{p}_{\mathcal{S}} + (1-\xi)\delta = \xi\delta + (1-\xi)\delta = \delta. \tag{53}$$

Thus:

$$\Pr\left[p_{\mathcal{X}^2} \leq \delta\right] \geq 1 - 2e^{-2S_\ell(1-\xi)^2\delta^2}, \tag{54}$$

which completes the proof. □

C.4 Proof of Proposition 4 (Stability of the mPL Audit)

Proposition 4 (Stability of the mPL audit)**.** *For a fixed sensitive position, let $p_0(\cdot \mid C)$ and $p_1(\cdot \mid C, \mathbf{Y})$ be the true pre-release prior and posterior marginals, and let $\widehat{p}_0, \widehat{p}_1$ be their fitted counterparts. Assume a finite candidate space, that all four distributions assign each candidate probability at least $\gamma > 0$, and that distinct candidates have distance at least $d_{\min} > 0$. Let n denote the number of auxiliary training records used to fit the joint prior from which $\widehat{p}_0$ and $\widehat{p}_1$ are derived. If $\mathbb{E}_{C,\mathbf{Y}}[\mathrm{KL}(p_0\|\widehat{p}_0) + \mathrm{KL}(p_1\|\widehat{p}_1)] \leq C_0 n^{-\alpha}$, then, for every fixed candidate pair (x_i, x_j),*

$$\mathbb{E}_{C,\mathbf{Y}}\left[\left|\mathrm{mPL}(x_i, x_j; C, \mathbf{Y}) - \widehat{\mathrm{mPL}}(x_i, x_j; C, \mathbf{Y})\right|\right] \leq \frac{2\sqrt{C_0}}{\gamma d_{\min}}\, n^{-\alpha/2}.$$

Proof. The log function is $1/\gamma$-Lipschitz on $[\gamma, 1]$. Applying this to the two posterior and two prior log terms, followed by the reverse triangle inequality, gives

$$|\mathrm{mPL} - \widehat{\mathrm{mPL}}| \leq \frac{\|p_0 - \widehat{p}_0\|_1 + \|p_1 - \widehat{p}_1\|_1}{\gamma d_{\min}}.$$

Pinsker's inequality and Jensen's inequality bound the expectation of the numerator by $\sqrt{2\mathbb{E}\mathrm{KL}(p_0\|\widehat{p}_0)} + \sqrt{2\mathbb{E}\mathrm{KL}(p_1\|\widehat{p}_1)} \leq 2\sqrt{C_0}\, n^{-\alpha/2}$. □

D Additional Details of Case Study

In this section, we provide additional details for the case study, including the design of the data perturbation and utility loss, serving as complementary material to Section 4 of the main paper.

D.1 2-Level Data Perturbation

Let $\mathcal{U}$ denote the complete set of word embeddings in the dataset. We define $\mathcal{X}_1$ and $\mathcal{X}_2$ as the subsets corresponding to PII and PoII embeddings, respectively. The overall set of sensitive embeddings is given by $\mathcal{X} = \mathcal{X}_1 \cup \mathcal{X}_2$, which is a subset of the full embedding set, i.e., $\mathcal{X} \subseteq \mathcal{U}$. To identify clear instances of PII, we apply *Named Entity Recognition (NER)* using the spaCy Python library[1]. Specifically, we assign a Level 1 privacy label to any word token identified by the NER model as a likely PII entity, such as `PERSON`, `GPE`, or `ORG`. These Level 1 tokens form the subset $\mathcal{X}_1 \subset \mathcal{U}$.

For the complementary set $\mathcal{U} \setminus \mathcal{X}_1$, we adopt the cosine-similarity-based approach proposed by Hassan et al. (Hassan et al., 2023) to identify *Potentially Identifiable Information (PoII)*. Specifically, we compute the cosine similarity between each token in $\mathcal{U} \setminus \mathcal{X}_1$ and the tokens in $\mathcal{X}_1$ using pretrained GloVe embeddings. The top 10% of tokens from $\mathcal{U} \setminus \mathcal{X}_1$ with the highest similarity scores are labeled as Level 2 privacy and form the PoII set $\mathcal{X}_2$. All remaining tokens, those not classified as either PII or PoII, are treated as non-sensitive.

D.2 Utility Loss Calculation

For a sensitive token x_i replaced by y_k, we define the replacement cost using normalized cosine distance:

$$\begin{aligned} c_{i,k} &= 1 - \frac{\mathrm{Sim}(x_i, y_k) + 1}{2} = \frac{1 - \mathrm{Sim}(x_i, y_k)}{2}, \\ \mathrm{Sim}(x_i, y_k) &= \frac{\mathbf{e}(x_i)^\top \mathbf{e}(y_k)}{\|\mathbf{e}(x_i)\|_2 \, \|\mathbf{e}(y_k)\|_2}. \end{aligned} \tag{55}$$

Here $\mathbf{e}(\cdot)$ denotes the token embedding. The cost lies in $[0, 1]$, with smaller values indicating greater semantic similarity. This same cost is used for parameter selection, Bayesian remapping, and utility evaluation.

During parameter selection, the expected loss averages $c_{i,k}$ over the mechanism's possible outputs and over sensitive positions, as in Eq. (10). For reported held-out utility, we first average realised replacement costs over sensitive positions within each record, then average over records containing sensitive positions. After remapping, the cost is evaluated at $f_t(y_k)$, where t is the source token's sensitivity tier.

E Additional Experimental Details and Results

E.1 Experiments Compute Resources

All experiments were conducted on a workstation equipped with an Intel Core i9-13900F processor with 24 cores and clock speeds ranging from 2.00 GHz to 5.60 GHz. The system has 32 GB of DDR5 memory running at 4800 MHz and an NVIDIA GeForce RTX 4090 GPU with 24 GB of GDDR6X VRAM. Runtime varied across datasets and attacker architectures. On a single RTX 4090

[1] spaCy: Industrial-Strength Natural Language Processing in Python. Available at: `https://spacy.io`

GPU, the largest attacker-training runs completed in approximately 2 minutes per run, while a full auditing-and-repair pipeline for one experimental setting required approximately 8–20 minutes.

E.2 Joint Attacker: Model, Training, and Inference

Joint model. The attacker models the ordered sensitive tokens $\mathbf{x} = (x_1, \ldots, x_L)$ conditional on public-token bag-of-words features of C. It combines a learned chain-potential model and a count-based Markov model:

$$\begin{aligned} \widehat{q}_N(\mathbf{x} \mid C) &= \sum_h \pi_{N,h}(C) \frac{\prod_{j=1}^{L} \phi_{h,j}(x_j) \prod_{j=1}^{L-1} \psi_h(x_j, x_{j+1})}{Z_h(L)}, \\ \widehat{q}_C(\mathbf{x} \mid C) &= \sum_h \pi_{C,h}(C) \mu_h(x_1) \prod_{j=1}^{L-1} T_h(x_{j+1} \mid x_j). \end{aligned} \tag{56}$$

The neural component learns context-dependent mixture gates, position-dependent unary potentials, and adjacent-token pairwise potentials. The count component estimates initial and transition probabilities with smoothing and backoff. Both share parameters across sensitive-sequence lengths. The final prior is $\widehat{q} = (1 - w)\widehat{q}_N + w\widehat{q}_C$.

Posterior inference. Index all chain components by r, with prior mixture weights $a_{N,h}(C) = (1 - w)\pi_{N,h}(C)$ and $a_{C,h}(C) = w\pi_{C,h}(C)$. For local release likelihoods $\ell_j(a) = \mathcal{M}_{g(a)}(y_j \mid a; \alpha_{g(a)})$, forward–backward inference computes each component's evidence e_r and posterior marginal $p_{r,\ell}$:

$$\begin{aligned} e_r &= \sum_{\mathbf{x}} q_r(\mathbf{x} \mid C) \prod_j \ell_j(x_j), \\ p_{r,\ell}(a) &= \frac{\sum_{\mathbf{x}: x_\ell = a} q_r(\mathbf{x} \mid C) \prod_j \ell_j(x_j)}{e_r}. \end{aligned} \tag{57}$$

The final posterior marginal is

$$\widehat{p}_{\boldsymbol{\alpha}}(X_\ell = a \mid C, \mathbf{y}) = \sum_r \frac{a_r(C) e_r}{\sum_{r'} a_{r'}(C) e_{r'}} p_{r,\ell}(a). \tag{58}$$

Thus mixture weights are updated using the observed release. Prior marginals use the same computation with $\ell_j = 1$. Multiplying the local release likelihoods into the unary factors preserves the chain structure, so component normalizers and marginals are computed by forward–backward recursions (Kschischang et al., 2001; Sutton & McCallum, 2012). For R chain components and at most K candidate states per position, dense inference takes $O(RLK^2)$ time, avoiding enumeration of the K^L joint configurations. The resulting marginals are exact for the fitted structured distribution before numerical protection, up to floating-point precision.

Training and mixture selection. The neural component is first trained by joint negative log likelihood and then fine-tuned using

$$\mathcal{L} = -\frac{1}{N} \sum_i \left[\log \widehat{q}_N(\mathbf{x}^{(i)} \mid C_i) + \frac{1}{L_i} \sum_j \omega_{ij} \log \widehat{q}_N(x_j^{(i)} \mid \mathbf{x}_{-j}^{(i)}, C_i) \right] + \mathcal{R}, \tag{59}$$

over records with $L_i > 0$. Here ω_{ij} are frequency-based weights derived from neighboring sensitive tokens in the training data, and $\mathcal{R}$ regularizes the chain potentials. Both stages use Adam and validation-based checkpoint selection. The count model is fitted using responsibility-weighted counts. After fitting both models, $w \in [0, 1]$ is selected by continuous minimization of validation joint negative log likelihood, including comparison with both endpoints. The resulting prior is fixed during mechanism selection and testing, with no additional temperature scaling.

Numerical protection. Inference uses scaled recursions and log-sum-exp arithmetic, with likelihoods and component marginals lower-bounded by 10^{-300} before taking logarithms. For the empirical mPL audit, each prior and posterior candidate probability receives an additive $\zeta = 10^{-8}$ before computing log odds, and the distance denominator is $d_{a,b} + 10^{-8}$.

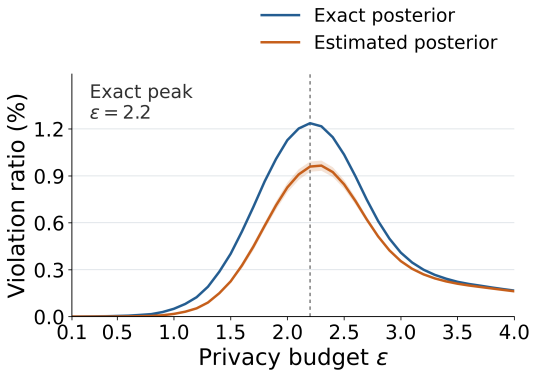

Figure 6: Fitted- and exact-posterior violation ratios for fixed EM across privacy budgets in the synthetic world.

E.3 mPL Distributions on the Three Text Datasets

To examine how budget violations are distributed beyond their aggregate frequency, Figure 7 extends the pre-remapping EM leakage example to all three datasets at $\epsilon = 2.5$. Most observed mPL values fall below the budget, while a small right tail crosses it in each dataset. Thus, budget violations coexist with a much larger population of observations whose leakage remains below the threshold. This pattern motivates examining the upper tail explicitly: the violation ratio captures events that a summary of typical leakage can obscure.

E.4 Detailed Synthetic Evaluation of Posterior Approximation

Because the true joint distribution is unknown for real-world data, we construct a synthetic world in which the exact prior and posterior are computable. Each record contains eight sensitive tokens and six public-context tokens governed by a shared latent topic, inducing within-record dependencies. The sensitive domain contains 100 candidates, the release space contains 500 tokens, and all tokens have 32-dimensional embeddings. Releases follow the same exponential mechanism used in the main framework.

Exact prior and posterior marginals are computed from the generator parameters using forward–backward inference within each topic and marginalization over topics. The fitted prior is learned from auxiliary data, and its posterior is obtained by combining it with the known release likelihood through Bayes' rule. Thus, the two evaluations use the same release mechanism but differ in whether the joint prior is known or fitted. We use 6,000 records for training, 2,000 for validation, and 1,000 for mechanism selection, and evaluate on a separate set of 40,000 records per seed. We compare EM and AmPL at $\epsilon \in \{2.2, 3.0, 3.5, 4.0\}$ and report means and sample standard deviations across seeds. Violation ratios are measured before remapping, while utility is reported before and after remapping. The generator parameters are used only for exact-posterior evaluation and are not provided to the fitted attacker or the deployed parameter-selection procedure.

Choice of privacy budgets. Figure 6 motivates the budgets used in Table 2. With EM fixed, the exact-posterior violation ratio peaks near $\epsilon = 2.2$, while the fitted-posterior curve follows the same non-monotone trend and peaks near $\epsilon = 2.3$. The selected budgets cover this peak and its subsequent decline; among them, $\epsilon = 2.2$ yields the highest EM violation ratio and therefore the most demanding observed-leakage comparison.

E.5 Effect of PoII Protection on PII Leakage.

Table 5 examines whether protecting PoII really helps control leakage about PII. AmPL-1 isolates the role of PoII protection by keeping the PII perturbation fixed and releasing PoII unperturbed. The PII violation ratio increases in every reported dataset–budget setting. At $\epsilon = 2.5$, it rises from 0.000041% to 0.001696% on AG News, from 0.000004% to 0.006817% on IMDB, and from 0.000039% to

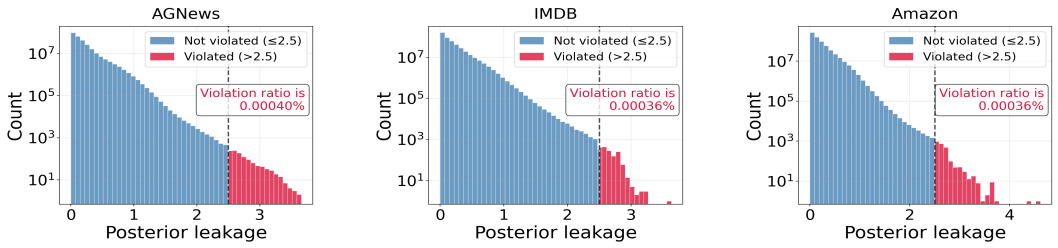

Figure 7: mPL distributions for EM on AG News, IMDB, and Amazon at $\epsilon = 2.5$.

Table 5: AmPL-1 (PoII-protection ablation).

Dataset	ϵ	PII violation ratio (%)		PII top-1 accuracy (%)	
		AmPL	AmPL-1	AmPL	AmPL-1
AGNews	2.4	0.000031 ± 0.000049	0.002329 ± 0.001457	22.76 ± 9.38	36.00 ± 5.24
AGNews	2.5	0.000041 ± 0.000038	0.001696 ± 0.000900	26.84 ± 8.71	37.19 ± 5.05
AGNews	2.6	0.000079 ± 0.000163	0.000962 ± 0.000904	22.66 ± 13.64	33.78 ± 9.31
IMDB	2.4	0.000004 ± 0.000009	0.014072 ± 0.021992	29.25 ± 5.67	29.32 ± 5.64
IMDB	2.5	0.000004 ± 0.000009	0.006817 ± 0.008352	27.99 ± 5.75	28.16 ± 5.61
IMDB	2.6	0.000014 ± 0.000031	0.004326 ± 0.005428	27.03 ± 8.14	27.13 ± 7.77
Amazon	2.4	0.000003 ± 0.000007	0.005713 ± 0.002840	48.27 ± 3.42	47.29 ± 2.44
Amazon	2.5	0.000039 ± 0.000070	0.003632 ± 0.001899	48.32 ± 4.82	47.50 ± 3.92
Amazon	2.6	0.000052 ± 0.000076	0.003328 ± 0.001887	50.89 ± 5.51	49.65 ± 4.71

0.003632% on Amazon. PII top-1 identification accuracy also increases on AG News from 26.84% to 37.19%, but remains similar on IMDB and Amazon. Because PII receives the same perturbation in both conditions, these results show that unprotected PoII supplies additional evidence about PII. Protecting PII and PoII together is therefore necessary to control leakage under joint observation.

E.6 Empirical PBmPL Thresholds

Table 6 summarizes the observed pre-remapping AmPL violation frequencies on the probability scale used by PBmPL. For each dataset and budget, we express the mean violation ratio across runs as a probability $\delta^\star$ and set $\widetilde{\delta} = 1.05\delta^\star$. The resulting empirical thresholds range from approximately 5.46×10^{-7} to 1.81×10^{-6} across the reported settings.

E.7 Utility Loss

To assess whether the utility benefit of remapping extends across datasets, Figure 8 extends the AG News utility comparison in the main text to IMDB and Amazon. Across all three datasets and the reported budgets, remapping lowers the realised loss for EM, AmPL, and AmPL-P. The consistent reduction shows that remapping improves semantic fidelity under the reported replacement-cost metric across these mechanisms and datasets.

Downstream classification utility. We also evaluate four-class AG News classification at $\epsilon = 2.5$ to complement replacement-cost measurements with task-level utility. For each run, a logistic-regression classifier on mean-pooled channel embeddings is trained on training set and evaluated on test set. We consider both training on clean inputs and training on a separately perturbed release of the training inputs. Table 7 reports accuracy and macro-F1. Remapping improves AmPL accuracy from 79.87% to 82.00% for clean-trained classifiers and from 81.00% to 82.27% for released-trained classifiers; macro-F1 improves in both settings as well. After remapping, AmPL accuracy is close to EM's 82.13% and 82.33%, respectively, while the clean-input reference achieves 83.60%. These results show that the utility gains from remapping also appear in downstream classification.

Table 6: Empirical PBmPL thresholds for the three text datasets.

ϵ	2.40	2.50	2.60
AG News	7.9997×10^{-7}	1.2099×10^{-6}	1.2727×10^{-6}
IMDB Review	6.9401×10^{-7}	5.4627×10^{-7}	6.3803×10^{-7}
Amazon Review	8.5693×10^{-7}	1.4796×10^{-6}	1.8073×10^{-6}

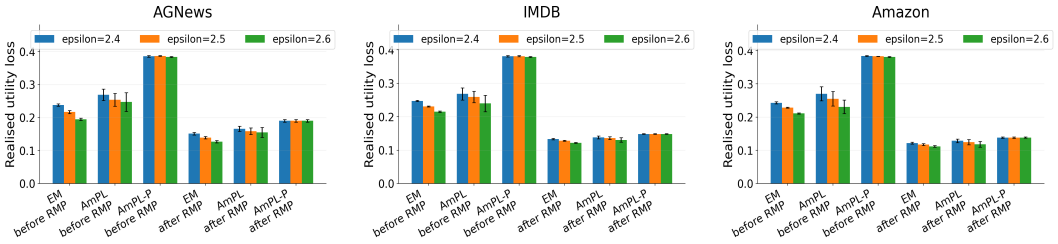

Figure 8: Utility loss before and after remapping on the three text datasets.

E.8 Tradeoff Between Utility and Violation Rate

To examine how the two perturbation coefficients affect the privacy–utility tradeoff, Figure 9 evaluates 100 combinations of α_1 and α_2 between 0.1 and 1.0 on each text dataset at $\epsilon = 2.5$. Both violation ratio and realised utility loss are measured before remapping, and the horizontal axis expresses the violation ratio as a fraction. Lower utility loss generally accompanies a higher violation ratio, while settings with similar violation ratios can differ substantially in utility loss. Thus, controlling violations still leaves meaningful scope for utility improvement through the choice of coefficients. This observation motivates including utility explicitly when selecting the perturbation strengths.

E.9 Posterior Leakage Comparison with Wider Privacy-Budget Range.

To examine leakage beyond the budgets emphasized in the main experiments, Figures 10 and 11 evaluate EM over a wider range of ϵ on all three text datasets. Mean mPL increases with ϵ, whereas the pre-remapping violation ratio peaks at intermediate budgets and then declines. These trends are compatible because the violation threshold also increases with ϵ: average leakage can rise while fewer observations exceed the corresponding budget. The two measurements therefore capture complementary aspects of the leakage distribution—its average magnitude and its frequency of exceeding the specified budget.

These curves also support the main-text evaluation at $\epsilon \in \{2.4, 2.5, 2.6\}$. The three budgets lie within the intermediate transition region, where EM's violation ratio is rising or near its peak, depending on the dataset. They provide closely spaced operating points for comparing privacy–utility behavior in a region where budget violations are already evident, while the wider sweep places these comparisons in context.

E.10 Effect of Attacker Training Data Size

To assess how the amount of auxiliary data affects attacker inference, Figure 12 varies the training subset from 10% to 100% while keeping the validation set, test set, and vocabulary fixed. Test releases are generated by EM at $\epsilon = 2.5$. Inference quality is measured by the mean cosine similarity between each true sensitive-token embedding and its posterior-mean embedding. Similarity generally improves as more auxiliary records are available, with a larger overall gain on AG News and smaller gains on IMDB and Amazon. Improvements also tend to diminish toward the largest subsets. These results show that additional auxiliary data improves the semantic fidelity of posterior-based estimates, with the size of the benefit varying across datasets.

Table 7: AG News downstream classification at $\epsilon = 2.5$. "Clean-trained" and "released-trained" indicate which inputs are used to train the classifier; both are evaluated on the corresponding test release.

	Clean-trained		Released-trained	
Test input	Accuracy	Macro-F1	Accuracy	Macro-F1
Clean	83.60 ± 2.25	83.68 ± 2.41	–	–
EM, pre-remap	80.60 ± 1.85	80.73 ± 2.12	81.47 ± 2.57	81.60 ± 2.59
EM, post-remap	82.13 ± 2.29	82.34 ± 2.39	82.33 ± 2.46	82.50 ± 2.54
AmPL, pre-remap	79.87 ± 2.51	79.97 ± 2.57	81.00 ± 2.48	81.10 ± 2.46
AmPL, post-remap	82.00 ± 2.40	82.22 ± 2.64	82.27 ± 2.15	82.48 ± 2.28

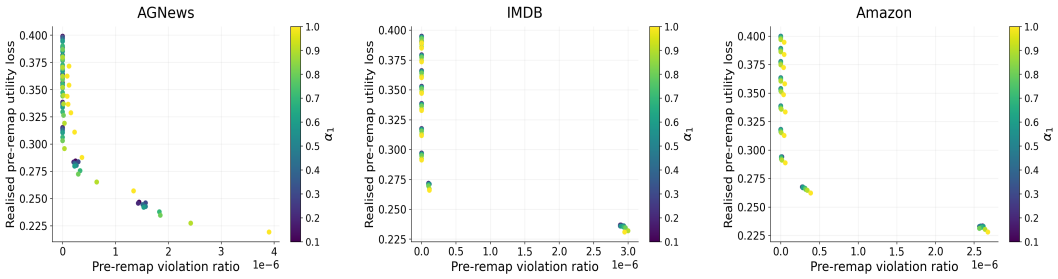

Figure 9: Privacy–utility trade-off across mechanism settings at $\epsilon = 2.5$.

E.11 Attacker Knowledge of the Embedding Model.

To examine the sensitivity of posterior-based leakage estimates to the attacker's channel knowledge, Table 8 compares matched and mismatched embedding-based release likelihoods on AG News. The learned prior, observed releases, and audit distance remain fixed, isolating the effect of the likelihood used in the Bayesian update. The mismatched likelihood yields slightly higher mean estimated mPL at all three budgets; at $\epsilon = 2.5$, the mean increases from 0.1830 to 0.1880. Both conditions retain the same increasing trend across budgets. A mismatched likelihood can assign more extreme relative probabilities to candidate secrets, producing larger estimated odds shifts without improving identification accuracy. Thus, the observed difference reflects the sensitivity of estimated mPL to the assumed release likelihood, while the underlying releases remain unchanged.

E.12 Generality Beyond Text.

To examine applicability beyond textual embeddings, we evaluate correlated numerical measurements from the same individual. This setting retains the central ingredients of the audit—a metric over candidate secrets and a prior-to-posterior update under joint release—while replacing tokens with quantized numerical blocks.

We use the Breast Cancer Wisconsin (Diagnostic) dataset (Wolberg et al., 1993), containing 569 patients and 30 continuous features. Each patient is represented by three protected ten-dimensional blocks: the mean, worst, and standard-error summaries, in that order. We fit feature standardization and an eight-centroid k-means codebook for each block using training patients only. Each secret is the index of its nearest centroid, and each exponential mechanism releases a centroid from the same block. The standardizer, codebooks, and block identities are public; patient identifiers, diagnosis labels, and original measurements are not supplied as attacker context. We specify a release policy assigning the worst block to tier 1 and the other two blocks to tier 2; this is a benchmark policy, not a clinical sensitivity annotation.

Table 9 reports pre-remapping violation ratios for the three privacy budgets. EM has mean estimated violation ratios of 1.20–1.23%, whereas AmPL reduces them to 0.00033–0.00065%. The reduction occurs at every reported budget, showing that AmPL also controls the observed budget violations in this setting with correlated numerical blocks. Together with the text experiments, these results

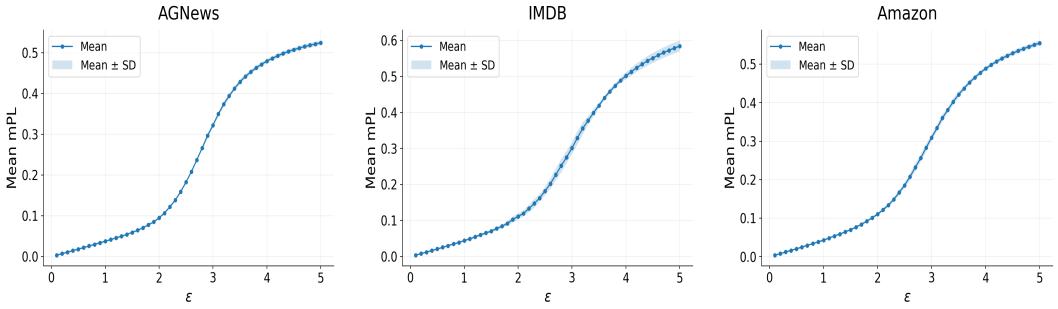

Figure 10: Mean mPL for EM across privacy budgets.

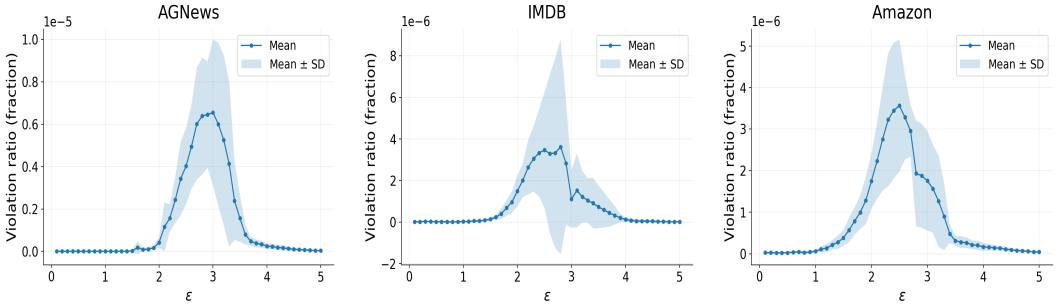

Figure 11: Violation ratio for EM across privacy budgets.

illustrate how the same metric-based audit and adaptive perturbation approach can be applied to different types of structured secrets.

F Licenses and Terms of Use for Existing Assets

We provide below a complete summary of all existing assets we used in this work, including citations, versions, URLs, and licenses.

Datasets

- **AG News** (Zhang et al., 2015). Used as a benchmark dataset. Available at: `https://www.kaggle.com/datasets/amananandrai/ag-news-classification-dataset`. License: For academic and research use only.
- **IMDB Movie Reviews** (Maas et al., 2011). Used as a benchmark dataset. Version 1.0. Available at: `http://ai.stanford.edu/~amaas/data/sentiment/`. License: For academic and research use only.
- **Amazon Reviews** (Zhang et al., 2015). Used as a benchmark dataset. Available at: `https://www.kaggle.com/datasets/kritanjalijain/amazon-reviews`. License: CC0 1.0 Universal.
- **Breast Cancer Wisconsin** (Wolberg et al., 1993). Used as a non-text tabular benchmark dataset for the beyond-text evaluation. Available at: `https://archive.ics.uci.edu/dataset/17/breast+cancer+wisconsin+diagnostic`. License: Creative Commons Attribution 4.0 International (CC BY 4.0).

Pre-trained Embeddings

- **GloVe** (Pennington et al., 2014). We use pre-trained GloVe word embeddings. Available at: `https://nlp.stanford.edu/projects/glove/`.

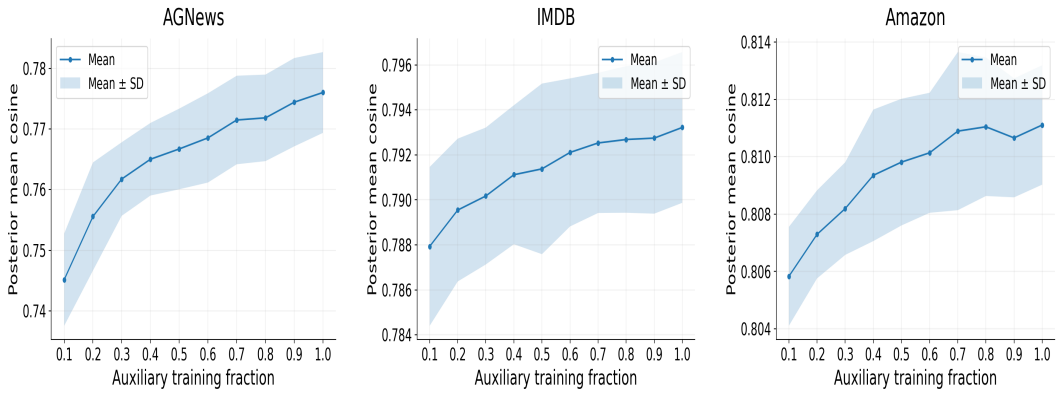

Figure 12: **Effect of auxiliary training set size on sensitive-token reconstruction.** The Joint attacker is refitted using nested subsets containing 10%–100% of the auxiliary training records. The validation set, test set, and vocabulary remain fixed. Test records are perturbed by EM at $\epsilon = 2.5$. Reconstruction is measured by the mean cosine similarity between each true sensitive-token embedding and the embedding averaged under the estimated posterior.

Table 8: Mean mPL on AG News with matched and mismatched assumed release channels.

	$\epsilon = 2.40$	$\epsilon = 2.50$	$\epsilon = 2.60$
matched	0.1593 ± 0.0031	0.1830 ± 0.0023	0.2086 ± 0.0015
mismatched	0.1652 ± 0.0021	0.1880 ± 0.0010	0.2125 ± 0.0020

Table 9: Violation ratio (%) on Breast Cancer Wisconsin (Diagnostic).

	$\epsilon = 0.10$	$\epsilon = 0.20$	$\epsilon = 0.30$
EM (mDP)	1.20373 ± 0.26419	1.22546 ± 0.21567	1.21737 ± 0.23636
AmPL	0.00033 ± 0.00073	0.00046 ± 0.00102	0.00065 ± 0.00146

Software Packages

- ▷ **spaCy**. Version 3.0. Used for PII identification. Available at: `https://spacy.io`. License: MIT License.

G Impact Statements

This work introduces mPL and PBmPL/AmPL to evaluate and reduce inferential privacy leakage when correlated releases are observed jointly. These tools can help identify leakage in representation-based ML applications. The same inference methods could also be used to attack deployed perturbation schemes, so their use calls for careful evaluation of the attacker and release setting.