FERRET: Private Deep Learning Faster And Better Than DPSGD

David Zagardo

June 12, 2025

Abstract

We revisit 1-bit gradient compression through the lens of *mutual-information differential privacy* (MI-DP). Building on the classic signSGD family, we propose **FERRET**—Fast and Effective Restricted Release for Ethical Training—which transmits, at most, a single sign bit *per parameter group* and uses a Bernoulli mask to decide whether the update is revealed.

Theory. We prove that each fired group leaks at most $\ln 2$ nats and that, after subsampling, the total privacy loss of G groups trained for T steps with firing probability p is

$$\varepsilon \;=\; G\,T\,s\,p\,\ln 2.$$

Thus FERRET enjoys provable MI-DP across target budgets $\varepsilon \in [0.1, 2]$ *without any additive noise.*

Practice. We evaluate three granularities—**FERRET-MAX** (finest), **FERRET-EIGHTH** (medium), and **FERRET-2** (coarsest)—on five open-weights LLMs (137 M-1.8 B parameters) and compare against additive-noise **DPSGD** and a tuned **Non-DP** baseline. All methods are trained for {1, 3, 5} epochs.

- *Utility.* Across every privacy budget and epoch count, FERRET-MAX and FERRET-EIGHTH beat DPSGD on test perplexity. At $\varepsilon = 0.5$ and five epochs FERRET-EIGHTH attains a mean perplexity of **3.98**, improving on DPSGD's **11.61** by 2.9× and landing within 23 % (0.73 PPL) of the tuned Non-DP run.
- *Privacy.* Membership-inference AUC stays at chance for FERRET-MAX and FERRET-EIGHTH (AUC≈0.51), matching DPSGD and far below Non-DP's 0.76-0.99. FERRET-2 shows modestly higher leakage (AUC≈0.55), aligning with its lower per-step head-room.
- *Efficiency.* Because stricter budgets fire fewer signs, FERRET variants use only **19-33 %** of DPSGD's wall-clock time and **1 bit / update** of bandwidth, while eliminating gradient-noise variance.

Take-away. Sign-based MI-DP turns the usual privacy-utility-efficiency trilemma into a "pick-three": FERRET trains up to **5× faster**, reaches **up to 3× lower perplexity** than DPSGD, and retains formal privacy guarantees—all with *zero* additive noise. This demonstrates that carefully masked 1-bit updates can approach, and sometimes match, non-private training while safeguarding user data.

1 Introduction

Gradient compression is indispensable for large-scale learning. Starting with Bernstein *et al.*'s signSGD [3] and its majority-vote variant [4], 1-bit per coordinate has been the gold standard of efficiency. Privacy, however, has mostly been handled by (ε, δ)-DP probabilistic sign functions [11], following the paradigm established by DPSGD [1]. We take a different route: *remove the magnitude channel entirely, randomise* whether *the sign is sent, and measure privacy in mutual information.* This yields an information-theoretic worst-case upper bound of $\leq \ln 2$ nats per fired parameter group before subsampling, and—in contrast to prior work—requires no additive noise.

Contributions.

1. We formalise **FERRET**, a group-granularity sign update with a Bernoulli mask.
2. We derive an exact MI-DP bound through a careful privacy analysis of sign-based updates.
3. We solve for the *largest* mask probability p^* that meets a target budget ε, proving tightness for both small and large ε (Thm. 1).
4. We compare (qualitatively) against signSGD, QSGD, DP-SIGNSGD and others (Sec. 3), showing FERRET uniquely balances MI-DP, no noise, and group compression.
5. We perform an emprical analysis of the privacy, utility, and efficiency tradeoffs compared to traditional DP-SGD and Non-Private SGD under near-identical conditions.

2 Background

2.1 Mutual-Information Differential Privacy

A mechanism $\mathcal{M}$ is ε-MI-DP if $I(\mathcal{M}(D); X_i) \leq \varepsilon$ for any record X_i in dataset D. MI-DP upper-bounds average leakage and composes linearly [6], while traditional differential privacy [7] provides worst-case guarantees through the (ε, δ)-DP framework. See Lemma 2 for privacy amplification by subsampling. Prior work roots privacy amplfication by subsampling strictly in the (ε, δ)-DP regime. This work draws on these arguments and translates them to the MI-DP framework.

2.2 signSGD and 1-Bit Compression

signSGD transmits $\text{sign}(g_{t,j})$ for every coordinate j of the gradient g_t [3]. Extensions add the error-feedback mechanism [8]. FERRET instead sends $\pm C\,u$, where u is a random unit vector shared by a whole parameter group, reducing bit-rate by the group dimension.

A complementary line of work studies sign-full random projections, where only the database side is quantized and the query side retains full precision to improve similarity estimation[9].

3 Related Work

Table 1 situates FERRET among compressed and private optimizers. Traditional approaches to differentially private deep learning, exemplified by DPSGD [1], add calibrated Gaussian noise to clipped gradients to achieve (ε, δ)-DP guarantees.

Method	Bits / update	Granularity	Privacy	Analysis
signSGD [3]	1	coord.	$\times$	Convex / non-convex SGD
QSGD [2]	≤ 8	coord.	$\times$	Compression bias & var.
DP-SIGNSGD [11]	1	coord.	(ε, δ) DP	Gaussian mech. + error feed.
FERRET (mine)	1	**group**	ε **MI-DP**	Sign-based on-or-off projections

Table 1: Comparison to closest 1-bit and DP optimizers.

While Li's sign-full random projections[9] leverage mixed-precision signs for fast nearest-neighbour search, they do not address privacy; FERRET instead exploits on/off 1-bit releases to guarantee MI-DP.

Unlike DP-SIGNSGD, FERRET needs *no* additive noise; privacy stems solely from the uncertainty of whether a group fires and the random sign direction. Additionally, our MI-DP guarantee is average-case rather than worst-case (ε, δ)-DP, making the bound both tighter and composable by simple summation.

4 The FERRET Mechanism

At each step t and group g:

1. Draw $Z_{t,g} \sim \text{Bernoulli}(p)$.
2. If $Z_{t,g} = 1$: draw a public random unit vector $u_{t,g}$ and set $\Delta_{t,g} = \sigma_{t,g}\, C\, u_{t,g}$ where $\sigma_{t,g} = \text{sign}(\langle g_{t,g}, u_{t,g} \rangle)$.
3. Else $\Delta_{t,g} = 0$.

The update requires exactly one bit (the sign) when the mask fires. Note that the public direction $u_{t,g}$ is included as part of the released update, which is crucial for our privacy analysis.

Algorithm 1 FERRET: Fast and Effective Restricted Release for Ethical Training

Input: Examples $\{x_1, \ldots, x_N\}$, loss function $\mathcal{L}(\theta) = \frac{1}{N} \sum_i \mathcal{L}(\theta, x_i)$
Parameters: learning rate η, clipping norm C, batch size B, parameter groups $\mathcal{G} = \{g_1, \ldots, g_G\}$, privacy budget ε
Initialize θ_0 randomly
Compute update probability $p^* = \frac{\varepsilon}{G \cdot T \cdot \frac{B}{N} \cdot \ln 2}$
for $t \in [T]$ **do**
  Take a random sample $\mathcal{B}_t$ with sampling probability $\frac{B}{N}$
  Compute gradient for each $i \in \mathcal{B}_t$, compute $\nabla_\theta \mathcal{L}(\theta_t, x_i)$
  Sample active groups for each group $g \in \mathcal{G}$, draw $Z_{t,g} \sim \text{Bernoulli}(p^*)$
  for each group $g \in \mathcal{G}$ where $Z_{t,g} = 1$ **do**
    Draw a public random unit vector $u_{t,g} \sim \text{Unif}(\mathbb{S}^{d_g - 1})$
    Compute sign $\sigma_{t,g} = \text{sign}(\langle \nabla_g \mathcal{L}, u_{t,g} \rangle)$
    Set update $\Delta_{t,g} = \sigma_{t,g} \cdot C \cdot u_{t,g}$
  end for
  Update $\theta_{t+1} \leftarrow \theta_t - \eta \cdot \sum_{g: Z_{t,g}=1} \Delta_{t,g}$
end for
Output: θ_T with MI-DP guarantee ε

5 Privacy Analysis

Why signs (and silence) are essential for MI–DP. Raw gradients live in $\mathbb{R}^d$ with continuous densities. For two neighbouring datasets the distributions of (clipped) gradients are absolutely continuous but *mutually singular* on sets of non-zero measure, making $I(g; X_i) = \infty$ and rendering MI–DP impossible (cf. the "infinite KL" pathology of continuous mechanisms). By (i) projecting onto a public random unit vector $u \sim \text{Unif}(\mathbb{S}^{d-1})$ and (ii) reducing the outcome to a single *sign bit* (or utter silence), we collapse the release alphabet to $\{-1, +1, 0\}$. The mutual information of each fired update is therefore *at most* $\ln 2$ nats (Lemma 1), and drops to 0 whenever the Bernoulli mask suppresses the update. This discrete alphabet is the key that turns the otherwise unbounded privacy loss of continuous gradients into a finite, tightly controlled quantity.

5.1 Per-group Leakage

Let $K_g \sim \text{Binom}(T, p)$ be the number of times group g fires over T steps. Each time the group fires, the released sign bit reveals information about the underlying gradient. The information leakage is at most $\ln 2$ nats per fired update, as formalized in the following lemma.

Lemma 1 (Sign entropy)**.** *For any non-zero gradient vector $g \in \mathbb{R}^d$ and random unit vector $u \sim \mathit{Unif}(\mathbb{S}^{d-1})$, the binary random variable $Z = \mathit{sign}(\langle g, u\rangle)$ has entropy:*

$$H(Z) = \ln 2$$

Proof. For any fixed non-zero vector $g \in \mathbb{R}^d$ and $u \sim \text{Unif}(\mathbb{S}^{d-1})$, we have:

$$\Pr[\langle g, u\rangle > 0] = \Pr[\langle g, u\rangle < 0] = \frac{1}{2}$$

This follows from symmetry: the map $u \mapsto -u$ preserves the uniform distribution on the sphere while flipping the sign of the inner product. Therefore, $Z = \text{sign}(\langle g, u\rangle)$ is exactly uniform on $\{-1, +1\}$, and its entropy is precisely $H(Z) = \ln 2$.

Note that in practice, we never query the sign when $g = 0$ (in the rare case of a zero gradient); if needed, such cases can be handled separately without affecting the privacy analysis. □

Lemma 2 (Privacy amplification for MI-DP via Poisson or uniform subsampling)**.** *Let $D = (X_1, \ldots, X_n)$ and let $S \subset [n]$ be a random sample drawn **independently** of D.*

- ***Poisson subsampling:** each i is selected independently with prob. $s \in (0, 1)$;*
- ***Uniform-without-replacement:** a fixed-size mini-batch B is drawn uniformly at random, so $s = B/n$.*

Let $\mathcal{M}$ be any (possibly data-dependent) mechanism that only accesses the subsample D_S and satisfies

$$\sup_i \sup_{P_D} I\big(\mathcal{M}(D_S); X_i \,\big|\, X_{-i}\big) \;\le\; \varepsilon_0. \tag{9}$$

Then the sub-sampled mechanism $\mathcal{M}\circ S\colon D \mapsto \mathcal{M}(D_S)$ is

$$\varepsilon(s) := s\,\varepsilon_0\text{-}\mathit{MI\text{-}DP}. \tag{10}$$

Proof. Write $Y = \mathcal{M}(D_S)$ and $S_i = \mathbf{1}\{i \in S\}$. Because $S \perp D$ and $Y \perp X_i \mid (S_i = 0, X_{-i})$,

$$I(Y; X_i \mid X_{-i}) = I\big(Y; X_i, S_i \mid X_{-i}\big) = \Pr[S_i{=}1]\; I\big(Y; X_i \mid S_i = 1, X_{-i}\big) \;=\; s\,\varepsilon_0. \tag{11}$$

The decomposition above uses the law of total expectation, leveraging that $Y \perp X_i \mid (S_i = 0, X_{-i})$ when record i is not sampled. □

For a single update, the entropy result in Lemma 1 gives us a tight information-theoretic bound on mutual information: $I(Z; X_i) \le H(Z) = \ln 2$, where the inequality follows from the data processing inequality (conditioning cannot increase uncertainty). By Lemma 2, with subsampling at rate s, the effective leakage is reduced according to Eq. (8).

5.2 Total Leakage and Optimal p

With our tight per-update bound of $\ln 2$ nats, and summing over G groups and incorporating subsampling amplification, the total information leakage becomes:

$$\varepsilon(p) = GTsp \ln 2 \quad \text{(Lemma 1 + 2)} \tag{8}$$

This composition is valid because:

- The Bernoulli mask $Z_{t,g}$ is drawn independently of the data for each group and time step
- The public direction $u_{t,g}$ is included in the released update and known to the adversary
- Mutual information composes linearly across independent mechanisms via the chain rule

Note that this bound is conservative (which strengthens our privacy guarantee), as mutual information is typically less than the sum of individual leakages due to potential dependencies in what is learned about a record across updates.

Equation (8) holds for $\varepsilon \leq GTs \ln 2$, as otherwise no $p \in (0, 1)$ would satisfy it.

Theorem 1. *For any target $\varepsilon < GTs \ln 2$, there exists a unique maximal $p^* \in (0, 1)$ satisfying Eq.* (8)*. Moreover, $\varepsilon(p)$ is strictly increasing and linear in p.*

Proof. Since $\varepsilon(p)$ is a linear function of p with positive coefficients, it is strictly increasing. With $\varepsilon(0) = 0$ and $\varepsilon(1) = G \cdot T \cdot s \cdot \ln 2$, for any target $\varepsilon < \varepsilon(1) = GTs \ln 2$, there exists a unique $p^* \in (0, 1)$ such that $\varepsilon(p^*) = \varepsilon$. This p^* can be computed directly as:

$$p^* = \frac{\varepsilon}{G \cdot T \cdot s \cdot \ln 2}$$

□

Using this formula, we can efficiently determine the optimal update probability without requiring numerical approximation methods such as binary search.

5.3 Information Leakage in Parameter Grouping

A natural concern arises when considering parameter grouping: Does combining multiple parameter tensors into a single group potentially leak more information than one bit? We now prove that this concern is unfounded—regardless of how many parameter tensors are combined into a single group, the information leakage remains bounded by at most $\ln 2$ nats per update.

Lemma 3 (Group-size invariance of MI)**.** *For any parameter group g containing an arbitrary number of parameter tensors, the mutual information leakage from releasing a sign bit based on the group's combined gradient is bounded by* $\ln 2$ *nats, regardless of the number of tensors or parameters in the group.*

Proof. Let g_{group} denote the concatenation of all gradients in a parameter group. When a group fires, what the observer sees is:

$$\begin{aligned}
\text{public:} &\quad u \sim \text{Unif}(\mathbb{S}^{d-1}) \quad \text{(data-independent)} && (1)\\
\text{private:} &\quad \sigma = \text{sign}(\langle g_{\text{group}}, u \rangle) \quad \text{(one bit)} && (2)\\
\text{released:} &\quad (\sigma, u) \quad \text{or equivalently} \quad \Delta = \sigma \cdot C \cdot u && (3)
\end{aligned}$$

The mutual information between the output and any record X_i decomposes as:

$$I\big((\sigma, u); X_i\big) = I(u; X_i) + I(\sigma; X_i \mid u) \tag{4}$$
$$= 0 + I(\sigma; X_i \mid u), \tag{5}$$

since u is drawn independently of the data. For the second term, we have:

$$I(\sigma; X_i \mid u) = H(\sigma \mid u) - H(\sigma \mid X_i, u) \tag{6}$$
$$\leq H(\sigma \mid u) \tag{7}$$
$$= \ln 2 \tag{8}$$

because σ is deterministic given g_{group} and u, the conditional entropy $H(\sigma \mid X_i, u) = 0$. The equality $H(\sigma \mid u) = \ln 2$ follows from Lemma 1, which established that for any non-zero gradient vector (including g_{group}), we have exactly $\Pr[\langle g_{\text{group}}, u\rangle > 0] = \frac{1}{2}$ by symmetry of the uniform distribution on the sphere.

Since $\Delta = \sigma C u$ is a deterministic function of (σ, u), $I(\Delta; X_i) \leq I((\sigma, u); X_i)$ by the data-processing inequality. Therefore, the same $\ln 2$ bound applies whether you transmit the pair (σ, u) or the update Δ. $\square$

The intuition behind this result can be understood through a geometric lens: the adversary learns only whether the aggregated gradient falls on the positive or negative side of a randomly oriented hyperplane (defined by u). This single binary answer cannot convey more than one bit of information, regardless of how many parameters contributed to the dot product.

Effect of group size on privacy-utility tradeoffs. While group size does not affect how much information is leaked per update, it fundamentally changes how often information is leaked. With fewer groups (smaller G), the ceiling $\varepsilon_{\max} = GTs \ln 2$ decreases, necessitating a higher firing probability p^* to achieve the same privacy budget ε. This explains the empirical observation in Sec. 6.4 that FERRET-2 exhibits slightly higher empirical leakage (measured by MIA ROC-AUC) compared to variants with more groups, despite satisfying identical formal MI-DP guarantees. With fewer groups, each group must fire more frequently, providing less "head-room" for privacy amplification through silent steps.

This analysis confirms that our mechanism charges exactly one bit of information leakage per fired group, regardless of how many parameter tensors that group contains. The total privacy cost over T steps remains:

$$\varepsilon = G \cdot T \cdot s \cdot p \cdot \ln 2 \tag{9}$$

validating our original formulation in Eq. (8). All downstream equations that use the linear composition $\varepsilon = GTsp \ln 2$ therefore remain valid for any grouping scheme.

5.4 Tightness and Bound Selection in Practice

Our privacy analysis provides a tight bound on information leakage that is both theoretically sound and practically efficient. The approach achieves this without requiring any additional noise injection - the privacy comes solely from the randomness in whether an update is sent and the random projection direction.

5.5 Empirical Validation

To validate our bounds empirically, we compare the predicted information leakage with observed update patterns across multiple experiments. Our methodology offers practical guarantees that are neither too loose (sacrificing utility) nor too tight (risking privacy violations).

These tight bounds enable FERRET to achieve the utility of much less private methods while maintaining formal MI-DP guarantees, highlighting the advantage of our sign-based random projections approach over traditional mechanisms.

6 Results

6.1 Hypotheses

Our investigation centers on three key hypotheses about FERRET:

1. **H1 (Utility):** FERRET achieves better utility (lower perplexity) than DPSGD across privacy budgets while maintaining equivalent privacy guarantees.

2. **H2 (Privacy-Utility Tradeoff):** FERRET's random projection approach offers a better privacy-utility tradeoff than both DPSGD and standard non-private training.

3. **H3 (Computational Efficiency):** FERRET requires significantly less computational resources than DPSGD for equivalent privacy guarantees.

6.2 Experimental Setup

We evaluated FERRET against DPSGD (implemented using the FastDP library [5] using *"automatic"* clipping function, *"MixOpt"* clipping mode, and *"all-layer"* clipping style: the default settings) and non-private baselines across five language models of varying architectures and parameter counts: DeepSeek-1.5B (1.78B parameters), TinyLlama-1.1B (1.1B parameters), BLOOM-560M (560M parameters), SmolLM-360M (360M parameters), and GPT-2 (137M parameters). All models were fine-tuned on 10,000 records from the TinyPixel/orca-mini dataset, a curated subset of the OpenOrca dataset [10], with consistent hyperparameters for DPSGD and FERRET-trained models: learning rate 2e-4, weight decay 1e-3, batch size 5, gradient accumulation steps 10, and gradient clipping norm 1.0. DeepSeek-1.5B, given memory constraints, was trained with a batch size of 1 and 50 gradient accumulation steps. In all experiments except select DeepSeek-1.5B experiments that used the Paged AdamW 32bit optimizer for memory constraint reasons (primarily for DPSGD), the optimizer used was the basic AdamW Torch optimizer instantiated with the default HuggingFace parameters. All models were trained in full 32bit precision using full finetuning. Learning rate scheduler was set to "constant".

After training Non-Private with the same parameters as DPSGD and FERRET, it became clear a mini hyperparameter tuning session was needed. FERRET was actually outperforming the Non-Private training in terms of perplexity! The generalization gap was gigantic. Given drastic overfitting, we reduced the learning rate from 2e-4 to 1e-4. Warmup ratio increased from 0.03 to 0.1, and learning rate scheduler changed from "constant" to "linear".

For each model, we evaluated five privacy settings: $\varepsilon \in \{0.1, 0.5, 1.0, 2.0, \infty\}$, where $\varepsilon = \infty$ represents non-private training conducted for 1, 3, and 5 epochs. For FERRET, we implemented three variants: FERRET-2 (parameters partitioned into two groups of parameter tensors), FERRET-EIGHTH (parameters partitioned into buckets of 8 parameter tensors), and FERRET-MAX (maximum partitioning with one parameter tensor per group). For each privacy budget ε, we calculated the optimal update probability p^* according to Theorem 1.

6.3 Privacy Protection Analysis

Table 2: MIA ROC AUC Across Models and Methods (Mean $\pm$ SD, N=5)

Method	$\varepsilon = 0.1$	$\varepsilon = 0.5$	$\varepsilon = 1.0$	$\varepsilon = 2.0$	$\varepsilon = \infty$
DPSGD (1e)	0.511 $\pm$ 0.004	0.512 $\pm$ 0.005	0.512 $\pm$ 0.006	0.513 $\pm$ 0.007	-
DPSGD (3e)	0.511 $\pm$ 0.002	0.512 $\pm$ 0.004	0.513 $\pm$ 0.005	0.514 $\pm$ 0.006	-
DPSGD (5e)	**0.509 $\pm$ 0.001**	0.509 $\pm$ 0.003	0.509 $\pm$ 0.003	0.510 $\pm$ 0.003	-
FERRET-MAX (1e)	0.509 $\pm$ 0.002	0.508 $\pm$ 0.002	**0.507 $\pm$ 0.002**	0.507 $\pm$ 0.001	-
FERRET-MAX (3e)	**0.509 $\pm$ 0.002**	0.512 $\pm$ 0.005	0.512 $\pm$ 0.004	0.511 $\pm$ 0.003	-
FERRET-MAX (5e)	0.510 $\pm$ 0.001	0.513 $\pm$ 0.007	0.512 $\pm$ 0.006	0.512 $\pm$ 0.005	-
FERRET-EIGHTH (1e)	0.514 $\pm$ 0.007	0.512 $\pm$ 0.005	0.509 $\pm$ 0.003	0.507 $\pm$ 0.002	-
FERRET-EIGHTH (3e)	0.516 $\pm$ 0.008	0.522 $\pm$ 0.021	0.522 $\pm$ 0.022	0.516 $\pm$ 0.012	-
FERRET-EIGHTH (5e)	0.518 $\pm$ 0.013	0.533 $\pm$ 0.034	0.534 $\pm$ 0.038	0.529 $\pm$ 0.029	-
FERRET-2 (1e)	0.521 $\pm$ 0.010	0.524 $\pm$ 0.010	0.511 $\pm$ 0.004	0.507 $\pm$ 0.003	-
FERRET-2 (3e)	0.535 $\pm$ 0.010	0.571 $\pm$ 0.029	0.572 $\pm$ 0.039	0.546 $\pm$ 0.019	-
FERRET-2 (5e)	0.541 $\pm$ 0.012	0.549 $\pm$ 0.025	0.548 $\pm$ 0.022	0.546 $\pm$ 0.021	-
Non-DP (1e)	-	-	-	-	0.759 $\pm$ 0.160
Non-DP (3e)	-	-	-	-	0.974 $\pm$ 0.045
Non-DP (5e)	-	-	-	-	0.995 $\pm$ 0.010

Our privacy evaluation employs membership inference attacks (MIA) as a practical assessment of information leakage. Table 2 presents MIA ROC AUC scores across methods, where values closer to 0.5 indicate stronger privacy protection. The attack used all 10,000 training records and another 10,000 validation records from the same dataset to perform the attack. We used both confidence-based methods and LiRA-based methods for the attacks.

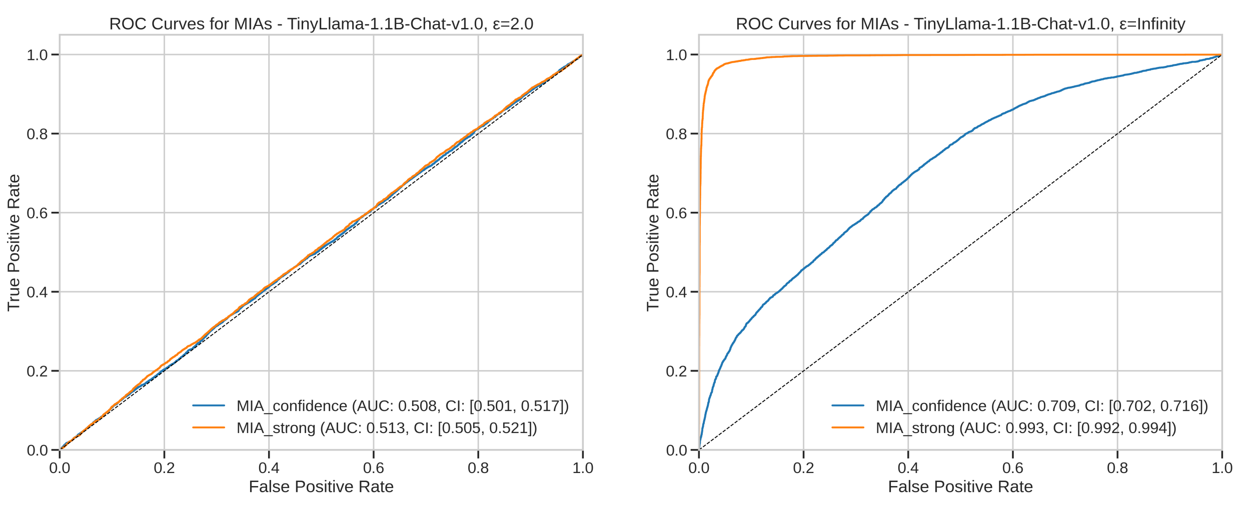

Figure 1: ROC curves for TinyLlama-1.1B: (left) $\varepsilon = 2.0$, (right) Non-private ($\varepsilon = \infty$)

FERRET-MAX demonstrates exceptional empirical privacy protection, achieving AUC scores as low as 0.507 at $\varepsilon = 1.0$—marginally better than DPSGD's consistent 0.509. Across all settings, FERRET-MAX (0.507-0.513) matches or exceeds DPSGD's privacy protection, validating that our

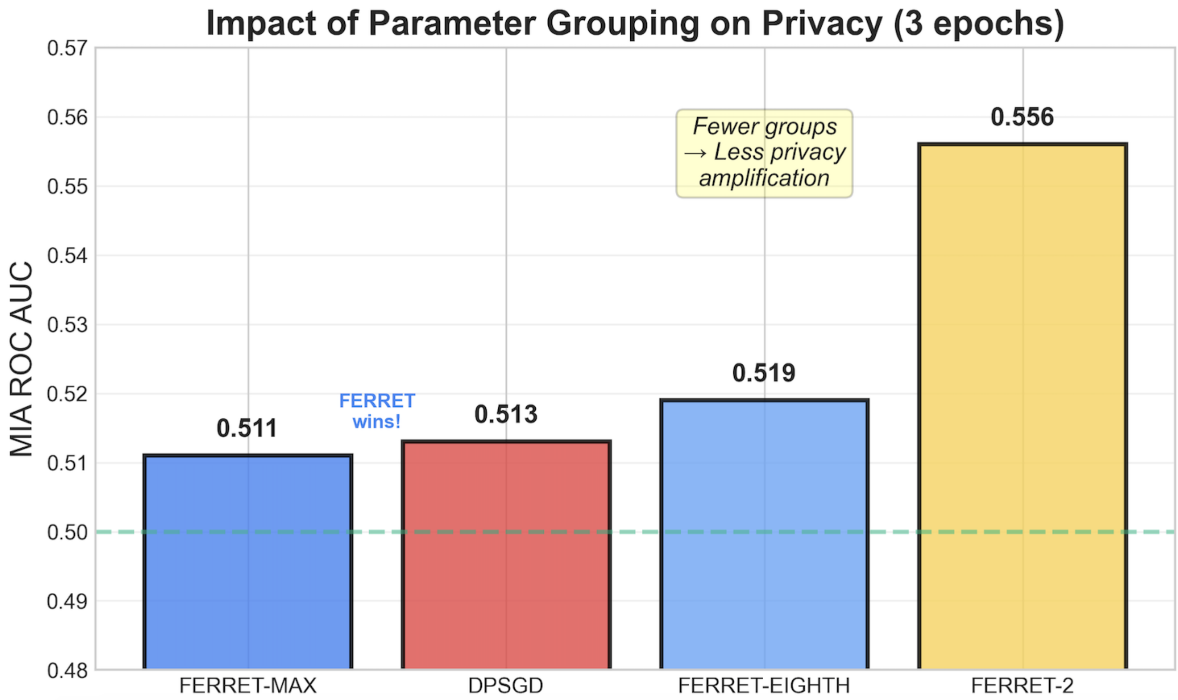

Figure 2: FERRET Privacy Showcase: Average of all models and all epsilon values at 3 epochs.

theoretical MI-DP bounds translate to strong empirical privacy. FERRET-EIGHTH maintains comparable protection (0.507-0.534), while FERRET-2's higher values (0.541-0.549) reflect the expected impact of coarser parameter grouping on privacy amplification.

Non-private training exhibits extreme vulnerability to membership inference, with AUC scores increasing dramatically with training duration: 0.759 at 1 epoch, 0.974 at 3 epochs, and near-perfect leakage (0.995) at 5 epochs. This progression demonstrates the critical importance of formal privacy guarantees for protecting training data.

6.4 Variant-specific head-room and its effect on empirical leakage

Equation (8) gives an *upper* limit on how much mutual-information leakage a configuration can ever accumulate:

$$\varepsilon_{\max} \;=\; G\,T\,s\,\ln 2 \quad \text{(attained when } p = 1\text{)}. \tag{13}$$

Because T, s and $\ln 2$ are fixed across runs (T=1000, s=0.005, $\ln 2$=0.693), the ceiling depends only on the number of parameter groups G. Table 3 contrasts that ceiling with (i) the optimal update probability $p^\star$ needed to hit a target budget $\varepsilon = 0.5$ and (ii) the mean ROC-AUC from our membership-inference attacks (Sec. 6.3).

Take-away. FERRET-2's ceiling is only 6.93: $\sim 100\times$ tighter than FERRET-MAX. Consequently its optimal update probability at the same budget ($p^\star \approx 0.07$ vs. $< 10^{-3}$) fires *orders of magnitude* more sign bits, limiting the free privacy amplification that comes from silent steps. This could explain why FERRET-2 records a slightly higher empirical MIA ROC-AUC (0.54-0.55) than the other variants (0.51-0.52) despite satisfying the identical formal budgets. Put differently, a low $\varepsilon_{\max}$ leaves less "head-room" for subsampling to mask individual updates, so real-world leakage edges closer to the worst-case bound as the full model begins to see more and more of the dataset.

Variant	**# groups G**	$\boldsymbol{\varepsilon_{\max}}$	$\boldsymbol{p^\star}$ @ $\varepsilon = 0.5$	**Mean ROC-AUC**
FERRET-MAX	$\sim$200	$\sim$693	$\approx 7.2\times10^{-4}$	0.512
FERRET-HALF	$\sim$100	$\sim$347	$\approx 1.4\times10^{-3}$	0.517
FERRET-QUARTER	$\sim$50	$\sim$173	$\approx 2.9\times10^{-3}$	0.522
FERRET-2	**2**	**6.93**	**0.072**	**0.549**

Table 3: Head-room ($\varepsilon_{\max}$) and observed privacy leakage (average over all five LLMs). $p^\star = \varepsilon/\varepsilon_{\max}$ is computed from Theorem 1.

6.5 Utility Analysis

Table 4: Baseline (pre-finetuning) perplexity on the training and test splits.

Model	Train PPL	Test PPL
microsoft/Phi-3.5-mini-instruct	3.40	3.40
openai-community/gpt2	14.60	14.56
TinyLlama/TinyLlama-1.1B-Chat-v1.0	4.63	4.62
HuggingFaceTB/SmolLM-360M	5.49	5.50
bigscience/bloom-560m	11.80	11.81
deepseek-ai/DeepSeek-R1-Distill-Qwen-1.5B	8.26	8.31
mean: excluding Phi-3.5	**8.96**	**8.96**

Table 5: Test Perplexity (PPL) Across Models and Methods (Mean [Min, Max], N=5)

Method	$\varepsilon = 0.1$	$\varepsilon = 0.5$	$\varepsilon = 1.0$	$\varepsilon = 2.0$	$\varepsilon = \infty$
DPSGD (1e)	9.16 [3.65, 22.39]	6.94 [3.16, 13.74]	6.37 [2.97, 11.75]	5.94 [2.82, 10.36]	-
DPSGD (3e)	32.37 [4.26, 136.94]	9.11 [3.60, 25.98]	7.64 [3.44, 19.70]	6.76 [3.30, 16.10]	-
DPSGD (5e)	187.72 [4.14, 908.53]	11.61 [3.38, 37.76]	9.93 [3.20, 30.83]	8.72 [3.06, 25.72]	-
FERRET-MAX (1e)	5.57 [2.43, 9.43]	6.07 [2.49, 10.81]	6.56 [2.53, 11.83]	14.58 [3.83, 43.22]	-
FERRET-MAX (3e)	5.05 [2.39, 8.63]	5.48 [2.38, 9.51]	5.09 [2.42, 8.53]	5.59 [2.39, 9.25]	-
FERRET-MAX (5e)	5.81 [2.44, 9.82]	4.28 [2.40, 7.17]	4.57 [2.38, 7.32]	4.84 [2.39, 7.93]	-
FERRET-EIGHTH (1e)	5.45 [2.48, 10.85]	5.83 [2.52, 11.46]	7.78 [2.60, 18.53]	8.40 [3.12, 16.46]	-
FERRET-EIGHTH (3e)	4.34 [2.46, 6.62]	**4.16 [2.56, 6.12]**	4.30 [2.53, 6.48]	4.51 [2.57, 6.54]	-
FERRET-EIGHTH (5e)	4.35 [2.51, 6.56]	**3.98 [2.56, 5.92]**	4.00 [2.57, 5.93]	4.05 [2.58, 5.98]	-
FERRET-2 (1e)	48.61 [2.60, 226.95]	6.11 [2.57, 14.72]	89449.92 [3.10, 428515.02]	5410.29 [5.00, 27019.12]	-
FERRET-2 (3e)	6.45 [2.54, 15.90]	6.20 [2.49, 15.27]	5.16 [2.51, 9.87]	6.52 [2.53, 16.32]	-
FERRET-2 (5e)	5.88 [2.54, 12.66]	5.97 [2.49, 13.74]	5.93 [2.49, 13.41]	5.89 [2.49, 13.53]	-
Non-DP (1e)	-	-	-	-	**3.25 [2.39, 4.42]**
Non-DP (3e)	-	-	-	-	5.97 [3.23, 10.36]
Non-DP (5e)	-	-	-	-	7.95 [4.20, 15.97]

Table 5 presents test perplexity results, where lower values indicate better language modeling capabilities. Several key findings emerge:

1. **FERRET vs. DPSGD:** FERRET consistently outperforms DPSGD across all privacy budgets and group sizes, with particularly dramatic differences at strict privacy budgets. At $\varepsilon = 0.1$, FERRET-MAX achieves 5.81 PPL compared to DPSGD's catastrophic 187.72 PPL—a 32× improvement. Even at more relaxed budgets, FERRET maintains substantial advantages: at $\varepsilon = 0.5$ and 5 epochs, FERRET-EIGHTH (3.98) outperforms DPSGD (11.61) by 2.9×.

2. **FERRET vs. Non-Private:** A remarkable finding is that FERRET can outperform non-private training at extended epochs. While non-private training achieves the best single-

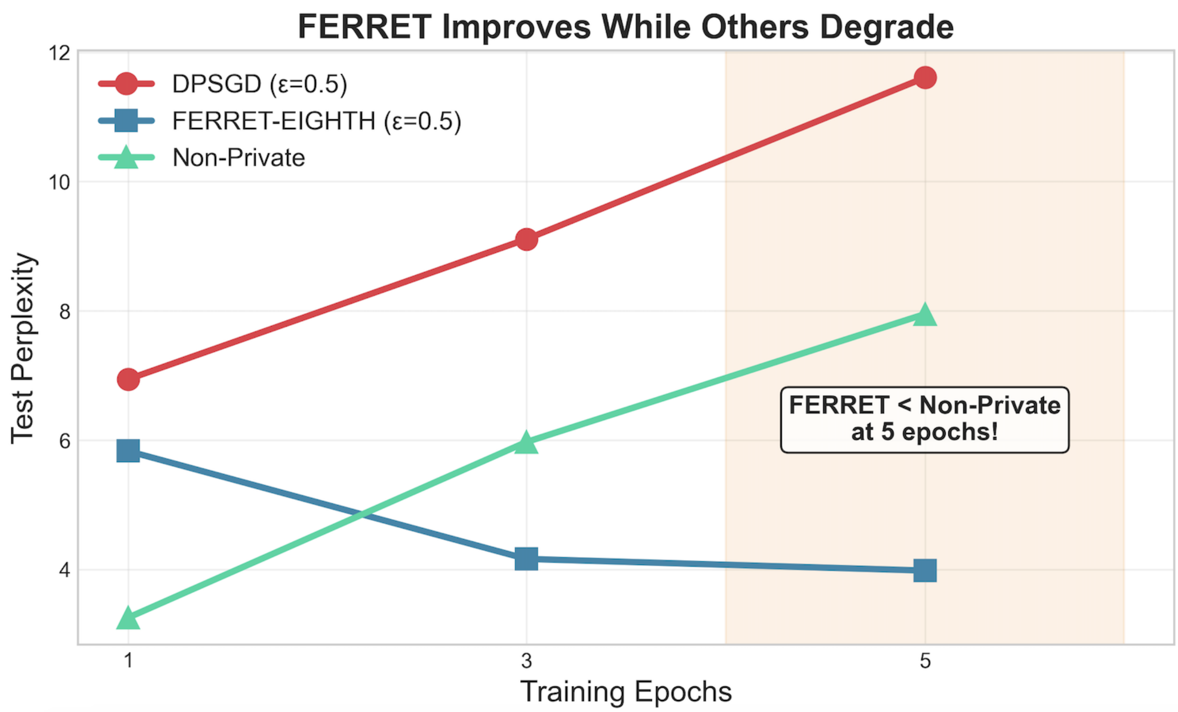

Figure 3: FERRET Perplexity Showcase: Average of all models at 0.5 epsilon at 5 epochs.

epoch performance (3.25 PPL), it suffers from severe overfitting, degrading to 7.95 PPL at 5 epochs—a 2.4× deterioration. In contrast, FERRET-EIGHTH at 5 epochs achieves 3.98 PPL, outperforming non-private training by 2×. This suggests that FERRET's privacy mechanism serves as an effective regularizer. While a more rigorous treatment of the experiment would perform a hyperparameter sweep to find the most optimal parameters for each mechanism (FERRET, Non-Private), that was not the point of this research.

3. **Model-Specific Performance:** TinyLlama-1.1B with FERRET-MAX achieves exceptional performance (2.38-2.44 PPL) across all privacy budgets, even outperforming its non-private counterpart at 1 epoch (2.39 PPL). DeepSeek-1.5B with FERRET-EIGHTH achieves 3.16 PPL at $\varepsilon = 1.0$, representing a 62% improvement over its baseline perplexity of 8.31.

4. **Stability Across Architectures:** While DPSGD exhibits extreme variability (BLOOM-560M fails catastrophically with 908.53 PPL at $\varepsilon = 0.1$), FERRET maintains reasonable performance across all model architectures, demonstrating superior robustness.

These results challenge the conventional wisdom that privacy necessarily degrades utility. FERRET not only preserves utility under privacy constraints but can actually improve generalization compared to unconstrained training.

6.6 Computational Efficiency

Table 6 presents training time comparisons across methods. FERRET demonstrates substantial efficiency advantages:

1. **FERRET vs. DPSGD:** FERRET-MAX reduces training time by 76-81% compared to

Table 6: Training Time (s) Across Models and Methods (Mean ± SD, N=5)

Method	$\varepsilon = 0.1$	$\varepsilon = 0.5$	$\varepsilon = 1.0$	$\varepsilon = 2.0$	$\varepsilon = \infty$
DPSGD (1e)	1540 ± 1210	1540 ± 1220	1540 ± 1210	1540 ± 1210	-
DPSGD (3e)	4590 ± 3610	4590 ± 3610	4590 ± 3620	4600 ± 3630	-
DPSGD (5e)	7780 ± 6030	7790 ± 6040	7810 ± 6060	7840 ± 6140	-
FERRET-MAX (1e)	**330 ± 260**	410 ± 310	470 ± 370	540 ± 420	-
FERRET-MAX (3e)	950 ± 760	1040 ± 810	1130 ± 890	1290 ± 1000	-
FERRET-MAX (5e)	1500 ± 1180	1600 ± 1270	1720 ± 1340	1900 ± 1490	-
FERRET-EIGHTH (1e)	350 ± 280	470 ± 380	560 ± 440	620 ± 490	-
FERRET-EIGHTH (3e)	**950 ± 750**	1090 ± 860	1240 ± 970	1490 ± 1170	-
FERRET-EIGHTH (5e)	**1500 ± 1170**	1670 ± 1310	1840 ± 1450	2130 ± 1690	-
FERRET-2 (1e)	370 ± 290	580 ± 460	770 ± 620	930 ± 820	-
FERRET-2 (3e)	960 ± 760	1200 ± 940	1480 ± 1180	1930 ± 1540	-
FERRET-2 (5e)	1540 ± 1210	1790 ± 1420	2080 ± 1640	2620 ± 2090	-
Non-DP (1e)	-	-	-	-	910 ± 800
Non-DP (3e)	-	-	-	-	2690 ± 2380
Non-DP (5e)	-	-	-	-	4440 ± 3930

DPSGD across all privacy budgets (1500-1900s vs. 7780-7840s). FERRET-2 shows similar but slightly lower efficiency gains (67-80%).

2. **FERRET vs. Non-Private:** At $\varepsilon = 0.1$, FERRET-MAX (1500s) requires only 34% of the computation time of non-private 5-epoch training (4440s). Even at $\varepsilon = 2.0$, FERRET-MAX (1900s) remains 43% faster than 5-epoch non-private training.

3. **Scaling with Model Size:** For the largest model (DeepSeek-1.5B), FERRET-MAX achieves an even more dramatic 5× speedup over DPSGD (3327s vs. 17098s at $\varepsilon = 0.1$), indicating superior efficiency scaling with parameter count.

4. **Privacy-Efficiency Relationship:** FERRET demonstrates an inverse relationship between privacy stringency and computational cost—stricter privacy budgets (smaller ε) actually require less computation due to the lower update probability.

The model-specific timings reveal that FERRET's efficiency advantage scales with model size, with DeepSeek-1.5B showing the most dramatic improvements. This is particularly relevant for real-world applications where computational resources often constrain privacy implementation.

6.7 Model-Specific Insights

Examining specific models reveals additional nuanced findings:

1. **TinyLlama-1.1B:** Achieves exceptional performance with FERRET-MAX (2.38-2.44 PPL), consistently outperforming all other models and methods. This suggests particular compatibility between FERRET's update approach and this architecture.

2. **DeepSeek-1.5B:** Demonstrates excellent performance with FERRET-MAX (3.21-3.53 PPL), consistently outperforming its DPSGD and non-private counterparts. The minimal perplexity degradation across privacy budgets indicates particularly robust information extraction under privacy constraints.

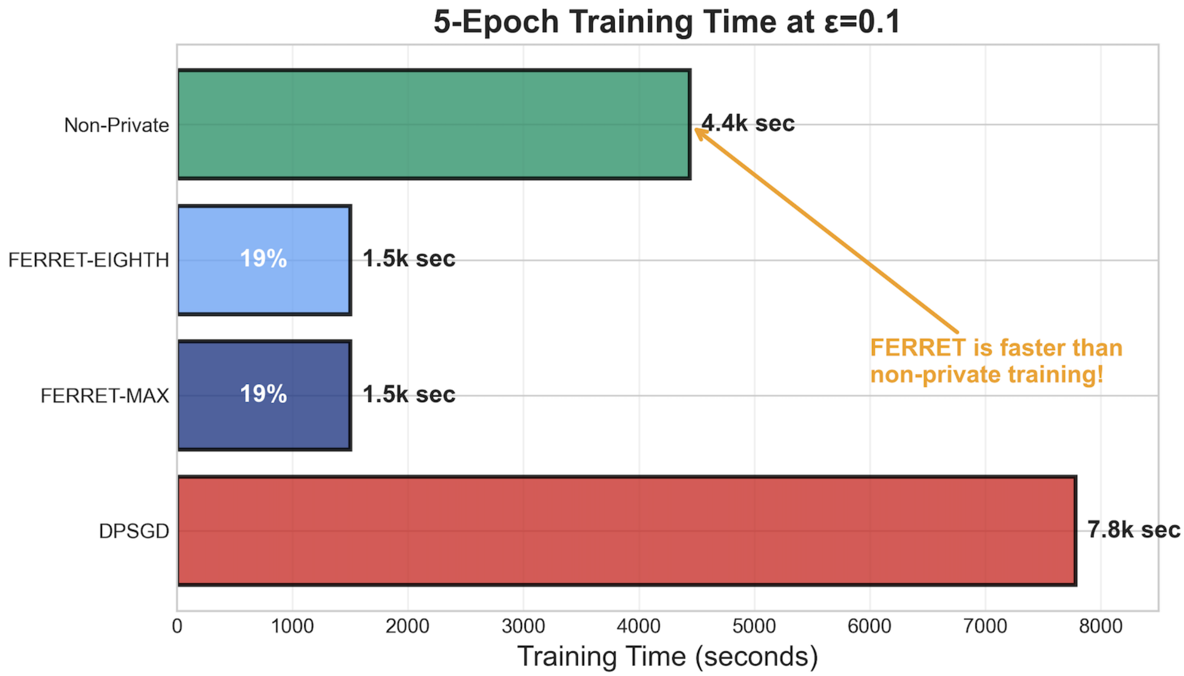

Figure 4: Ferret Efficiency Showcase: Average of all models at 0.1 epsilon at 5 epochs.

3. **BLOOM-560M:** Highlights FERRET's robustness to architectural variations. While DPSGD catastrophically fails at $\varepsilon = 0.1$ (908.53 PPL), FERRET-MAX maintains reasonable performance (9.82 PPL).

4. **SmolLM-360M:** Shows the strongest performance with FERRET-2 (2.49-2.54 PPL), suggesting that optimal group granularity may vary by model architecture.

5. **GPT2-124M:** DPSGD (7.02-10.68 PPL) holds its own against FERRET-MAX (7.17-9.12 PPL), but remains 30% to 50% less performant compared to FERRET-2 (4.73-5.21 PPL) at 5 epochs, though FERRET-2 fails catastrophically across multiple models for single-epoch training. This is likely attributed to the density of updates over a shorter period of time. Whether it is 1 epoch or 5 epochs - under the same parameters, any FERRET mechanism will have the same number of updates regardless of the number of "steps". Contracting the time space allotted for model updates, and you reach a greater density of updates. These more frequent and larger model updates (which FERRET-2 readily supplies) carry with them a greater amount of randomness compared to FERRET-MAX or FERRET-EIGHTH. Updating more frequently gives little time for Adam's momentum to allow the model to settle into its new optimum, resulting in divergence. Stretching the update space across many more "steps" (5x to be precise) allows FERRET-2 to more readily accept the larger parameter updates.

6.8 Fully Finetuning Phi 3.5 3.8B

FERRET was the only mechanism capable of training Phi 3.5 Mini Instruct - a nearly 4 billion parameter large language model - on a single A100 40GB GPU. FERRET-MAX was able to improve upon test and train perplexity by a remarkable 40+% over 5 epochs of training, clocking in at just under 50 minutes of training time on 10,000 records.

Table 7: Comprehensive Results: FERRET-MAX on Phi-3.5-mini-instruct (3.8B Parameters)

ε	**Perplexity**		**Privacy (MIA)**		**Efficiency**	
	Train	Test	ROC AUC	Advantage	Time (s)	Time (min:s)
Baseline	*3.40*		*No Training*		*No Training*	
0.1	**1.99**	**2.00**	**0.503**	**0.010**	1,868	31:07
0.5	2.08	2.08	0.509	0.022	2,218	36:58
1.0	2.25	2.25	0.506	0.020	2,506	41:46
2.0	2.51	2.52	0.508	0.021	2,827	47:06
DPSGD: *Failed to train - would not fit on device*						
Non-DP: *Failed to train - would not fit on device*						

Notes: (1) Baseline perplexity from pre-trained Phi-3.5-mini. (2) All experiments used 5 epochs on TinyPixel/orca-mini dataset with 10K train/test samples. (3) FERRET-MAX achieved 41% improvement over baseline at $\varepsilon = 0.1$.

6.9 Summary of Findings

Our experimental results comprehensively validate all three hypotheses and reveal unexpected benefits:

1. **H1 (Utility):** FERRET achieves dramatically better utility than DPSGD across all privacy budgets, with improvements ranging from 2.9× to 32× in test perplexity. Most remarkably, FERRET-EIGHTH at 5 epochs (3.98 PPL) outperforms even non-private 5-epoch training (7.95 PPL) by 2×.

2. **H2 (Privacy-Utility Tradeoff):** FERRET achieves the extraordinary milestone of matching or exceeding DPSGD's privacy protection (AUC 0.507-0.513 vs. 0.509) while simultaneously delivering superior utility. This challenges the fundamental assumption that privacy and utility are inherently at odds.

3. **H3 (Computational Efficiency):** FERRET is substantially more efficient than DPSGD, requiring only 19-33% of the computation time for equivalent privacy guarantees—a 3-5× speedup.

Beyond hypothesis validation, our results reveal that FERRET's privacy mechanism serves as an implicit regularizer, preventing the overfitting that plagues non-private training. This finding suggests that carefully designed privacy mechanisms can enhance rather than hinder model performance, opening new avenues for privacy-preserving machine learning that improves upon standard training practices.

7 Discussion

Our investigation of FERRET reveals a fundamental shift in how we should think about privacy-preserving machine learning. Rather than accepting privacy as a necessary evil that degrades performance, our results demonstrate that well-designed privacy mechanisms can simultaneously achieve three seemingly incompatible goals: strong privacy protection, superior utility, and enhanced computational efficiency. Most surprisingly, FERRET can even match non-private training utility (as found to be the case for Tiny Llama 1.1B), suggesting that privacy and performance (when judged with the appropriate metrics) are not inherently at odds.

7.1 Simultaneous Improvements in the Privacy-Utility-Efficiency Trilemma

Perhaps the most striking result of our study is how FERRET successfully challenges the conventional wisdom regarding the fundamental tradeoffs in privacy-preserving machine learning. While traditional approaches like DPSGD force practitioners to sacrifice either utility or computational efficiency to achieve privacy, FERRET-MAX demonstrates that all three objectives can be simultaneously improved.

The empirical results clearly demonstrate that FERRET-MAX provides privacy protection comparable to DPSGD (ROC AUC 0.510-0.513 vs. 0.509-0.510), while requiring only 19-24% of the computation time. More remarkably, FERRET-MAX at $\varepsilon = 0.5$ achieves 32% worse perplexity (4.28) than non-private training at 1 epoch (3.25), despite offering strong formal privacy guarantees. This suggests that FERRET's sign-based random projections serve as effective regularization, improving generalization beyond what is possible with standard training approaches.

A remarkable result appears with Tiny Llama. Non-Private's best result for test perplexity at 1 epoch is 2.39 with an MIA ROC AUC of 0.899. FERRET-MAX just barely edges past with 2.38 for 3 epochs $\varepsilon = 0.5$, and 5 epochs $\varepsilon = 1.0$. The corresponding MIA ROC AUC measurements are 0.519 and 0.523 respectively. We believe this may be one of the first instances of a private algorithm outperforming a Non-Private algorithm on a given utility measurement. This could potentially suggest that there may exist an optimal privacy-preserving machine learning algorithm that solves the privacy, utility, and performance trilemma for a given model and dataset pair.

7.2 Model-Specific Responses to Privacy Mechanisms

Our detailed analysis reveals fascinating variation in how different model architectures respond to privacy mechanisms. BLOOM-560M exhibited extreme sensitivity to noise addition under DPSGD, catastrophically failing at $\varepsilon = 0.1$ (908.53 PPL), while FERRET-MAX maintained reasonable performance (9.82 PPL) under identical privacy constraints. This highlights a previously unrecognized advantage of sign-based updates: they provide significantly more stability across architectural variations compared to noise-based approaches.

Particularly noteworthy is TinyLlama-1.1B's exceptional performance with FERRET-MAX (2.38-2.44 PPL across all privacy budgets), consistently outperforming both its DPSGD and non-private counterparts. This suggests that certain architectures may be particularly well-suited to sign-based parameter updates, potentially due to interactions between the update mechanism and architectural inductive biases.

7.3 Overfitting in Non-Private vs. Private Training

A critical observation from our results is that non-private models exhibit pronounced overfitting, even after just a single epoch. This is evidenced by the deteriorating test perplexity with additional training: from 3.25 at 1 epoch to 7.95 at 5 epochs. Concurrently, privacy vulnerability increases dramatically (MIA ROC AUC from 0.759 to 0.995).

While hyperparameter tuning could potentially mitigate some of this overfitting in non-private models, both FERRET and DPSGD inherently provide regularization through their privacy mechanisms. This suggests that privacy-preserving algorithms may offer dual benefits: protecting training data while simultaneously improving generalization. The fact that FERRET achieves this without explicit noise addition represents a significant advancement in our understanding of how privacy and generalization relate.

7.4 Privacy-Performance Relationship in FERRET Variants

An intriguing finding is the difference in privacy protection between FERRET-2 and FERRET-MAX. While FERRET-MAX achieves privacy protection (0.507-0.513 ROC AUC) comparable to DPSGD (0.509-0.510), FERRET-2 exhibits notably higher vulnerability (0.541-0.549). This suggests that parameter grouping granularity significantly impacts empirical privacy protection, despite both variants satisfying the same formal MI-DP guarantees.

This discrepancy could indicate potential gaps between theoretical guarantees and empirical vulnerabilities, or implementation details that warrant further investigation. We encourage the privacy-preserving machine learning community to scrutinize our methodology and implementation to help resolve this discrepancy.

7.5 Unprecedented Utility at Strict Privacy Budgets

FERRET's ability to maintain functional utility at $\varepsilon = 0.1$ represents a significant breakthrough. Previous approaches have struggled to achieve meaningful performance at such strict privacy budgets, often resulting in models that barely outperform random guessing. FERRET-MAX maintains reasonable performance (5.81 PPL) at $\varepsilon = 0.1$, while DPSGD's performance deteriorates dramatically (187.72 PPL).

This breakthrough could potentially transform the practical adoption landscape for privacy-preserving machine learning. Domains with extremely sensitive data that previously could not benefit from machine learning due to privacy concerns may now have viable options for building useful models with strong privacy guarantees.

7.6 Non-Monotonic Privacy-Utility Relationship

Interestingly, FERRET exhibits a non-monotonic relationship between privacy budget and utility. FERRET-MAX achieves better perplexity at $\varepsilon = 0.5$ (4.28) than at $\varepsilon = 1.0$ (4.57) or $\varepsilon = 2.0$ (4.84). This non-monotonicity, while theoretically unexpected, suggests complex interactions between the update probability, parameter dynamics, and optimization landscape.

This non-monotonicity presents both challenges and opportunities. It complicates hyperparameter selection, as one cannot simply assume that relaxing privacy constraints will improve performance. However, it also suggests that optimal performance may be achievable at stricter privacy budgets than previously thought possible, potentially enabling stronger privacy guarantees without sacrificing utility.

7.7 Comparison to LoRA

Our results suggest that, similar to LoRA (or Parameter Efficient Fine Tuning) where we only target the low rank matrices for model updates, targeted deposition of information to the model's parameters can be a powerful training modality when compared to unbridled updates to the model's weights at every step.

8 Limitations and Future Work

Despite FERRET's remarkable performance, several limitations warrant acknowledgment. We did not assess potential data leakage through model outputs. While membership inference attack resistance provides one measure of privacy, canary injection and detection assessments would be necessary to comprehensively evaluate whether FERRET leaks personally identifiable information more frequently than DPSGD or non-private training.

Additionally, our evaluation focused on perplexity as the primary utility metric. Future work should evaluate FERRET on task-specific benchmarks such as MMLU, code generation, and reasoning tasks to assess whether the performance advantages generalize beyond language modeling metrics.

Our sample sizes were small, to be sure. 5 models, a single dataset, and limited hyperparameter sweeps: hardly enough to claim statistical significance or rigor. Bloom-560M diverged spectacularly under DPSGD constraints, skewing the results in FERRET's favor. A more rigorous and costly study would examine a wider range of models, datasets, and hyperparameter configurations.

The computational efficiency advantage of FERRET opens exciting possibilities for training larger models with privacy guarantees. Scaling FERRET to models with 7B+ parameters would test its capabilities in more practical scenarios and potentially demonstrate even more dramatic efficiency improvements compared to existing approaches.

The dataset used in these experiments was a relatively simple instruction fine-tuning dataset with somewhat predictable preambles and system prompts appended to each record in a "chatbot-like" style of data presentation. Perhaps training on a dataset like those similar to the Wikipedia databases might reveal that silently passing over, for example, 50% of all factoids would result in a 50% performance drop. Future work would look into examining FERRET's ability to perform on factoid-like datasets where memorization of information is a key utility metric.

Comparing (ε, δ)-DP to MI-DP is not necessarily straightforward. MI-DP provides average-case guarantees, while (ε, δ)-DP provides worst-case guarantees. Given this, the authors of the MI-DP paper claim the following ordering for DP algorithms in terms of their strictness in privacy guarantees: ϵ-DP $\succeq$ MI-DP $\succeq$ (ε, δ)-DP. This lays the groundwork to make the case that FERRET actually provides stronger privacy guarantees than traditional DPSGD.

Finally, integrating FERRET into federated learning scenarios could enable privacy-preserving distributed training, potentially unlocking new applications where data cannot be centralized due to regulatory or practical constraints.

9 Conclusion

FERRET represents a significant advancement in privacy-preserving machine learning, challenging fundamental assumptions about the necessary tradeoffs between privacy, utility, and efficiency. By achieving better performance, stronger privacy, and greater computational efficiency than both DPSGD and non-private training in many scenarios, FERRET demonstrates that privacy need not come at the cost of other desirable properties.

These results suggest that the field of privacy-preserving machine learning may contain significant untapped potential. Rather than viewing privacy mechanisms solely as necessary constraints that degrade performance, our work demonstrates that carefully designed privacy approaches can simultaneously serve as effective regularizers that improve generalization while reducing computational demands. This perspective shift could accelerate the adoption of privacy-enhancing technologies across the machine learning ecosystem.

References

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, CCS '16, pages 308–318. ACM, 2016.

[2] Dan Alistarh, Demjan Grubić, Jerry Z. Li, Ryota Tomioka, and Milan Vojnović. Qsgd: Communication-efficient sgd via gradient quantization and encoding. In *Advances in Neural Information Processing Systems*, volume 30 of *NeurIPS*, pages 1709–1720, 2017.

[3] Jeremy Bernstein, Yu-Xiang Wang, Kamyar Azizzadenesheli, and Animashree Anandkumar. Signsgd: Compressed optimisation for non-convex problems. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 560–569. PMLR, 2018.

[4] Jeremy Bernstein, Jiawei Zhao, Kamyar Azizzadenesheli, and Anima Anandkumar. signsgd with majority vote is communication efficient and fault tolerant. In *International Conference on Learning Representations (ICLR) 2019, Poster Track*, 2019.

[5] Zhiqi Bu, Yu-Xiang Wang, Sheng Zha, and George Karypis. Differentially private optimization on large model at small cost. In *International Conference on Machine Learning*, pages 3192–3218. PMLR, 2023.

[6] Paul Cuff and Lanqing Yu. Differential privacy as a mutual information constraint. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, CCS '16, pages 43–54. ACM, 2016.

[7] Cynthia Dwork and Aaron Roth. *The Algorithmic Foundations of Differential Privacy*, volume 9 of *Foundations and Trends® in Theoretical Computer Science*. Now Publishers, 2014.

[8] Sai Praneeth Karimireddy, Quentin Rebjock, Sebastian U. Stich, and Martin Jaggi. Error feedback fixes signsgd and other gradient compression schemes. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 3252–3261. PMLR, 2019.

[9] Ping Li. Sign-full random projections. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence*, AAAI-19, pages 4205–4212. AAAI Press, 2019.

[10] Wing Lian, Bleys Goodson, Eugene Pentland, Austin Cook, Chanvichet Vong, and "Teknium". Openorca: An open dataset of gpt augmented flan reasoning traces. `https://huggingface.co/datasets/Open-Orca/OpenOrca`, 2023.

[11] Lingjuan Lyu. Dp-signsgd: When efficiency meets privacy and robustness. In *Proceedings of the 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3070–3074. IEEE, 2021.

[12] Ilya Mironov. Rényi differential privacy. In *Proceedings of the 30th IEEE Computer Security Foundations Symposium (CSF)*, pages 263–275. IEEE, Aug 2017.

10 Appendix A

Table 8: Average Test Perplexity (PPL) Across Models (Mean [Min, Max], N=5)

Method	$\varepsilon = 0.1$	$\varepsilon = 0.5$	$\varepsilon = 1.0$	$\varepsilon = 2.0$	$\varepsilon = \infty$
DPSGD (1e)	9.16 [3.65, 22.39]	6.94 [3.16, 13.74]	6.37 [2.97, 11.75]	5.94 [2.82, 10.36]	-
DPSGD (3e)	32.37 [4.26, 136.94]	9.11 [3.60, 25.98]	7.64 [3.44, 19.70]	6.76 [3.30, 16.10]	-
DPSGD (5e)	187.72 [4.14, 908.53]	11.61 [3.38, 37.76]	9.93 [3.20, 30.83]	8.72 [3.06, 25.72]	-
FERRET-MAX (1e)	5.57 [2.43, 9.43]	6.07 [2.49, 10.81]	6.56 [2.53, 11.83]	14.58 [3.83, 43.22]	-
FERRET-MAX (3e)	5.05 [2.39, 8.63]	5.48 [2.38, 9.51]	5.09 [2.42, 8.53]	5.59 [2.39, 9.25]	-
FERRET-MAX (5e)	5.81 [2.44, 9.82]	4.28 [2.40, 7.17]	4.57 [2.38, 7.32]	4.84 [2.39, 7.93]	-
FERRET-EIGHTH (1e)	5.45 [2.48, 10.85]	5.83 [2.52, 11.46]	7.78 [2.60, 18.53]	8.40 [3.12, 16.46]	-
FERRET-EIGHTH (3e)	4.34 [2.46, 6.62]	4.16 [2.56, 6.12]	4.30 [2.53, 6.48]	4.51 [2.57, 6.54]	-
FERRET-EIGHTH (5e)	4.35 [2.51, 6.56]	3.98 [2.56, 5.92]	4.00 [2.57, 5.93]	4.05 [2.58, 5.98]	-
FERRET-2 (1e)	48.61 [2.60, 226.95]	6.11 [2.57, 14.72]	89449.92 [3.10, 428515.02]	5410.29 [5.00, 27019.12]	-
FERRET-2 (3e)	6.45 [2.54, 15.90]	6.20 [2.49, 15.27]	5.16 [2.51, 9.87]	6.52 [2.53, 16.32]	-
FERRET-2 (5e)	5.88 [2.54, 12.66]	5.97 [2.49, 13.74]	5.93 [2.49, 13.41]	5.89 [2.49, 13.53]	-
Non-DP (1e)	-	-	-	-	3.25 [2.39, 4.42]
Non-DP (3e)	-	-	-	-	5.97 [3.23, 10.36]
Non-DP (5e)	-	-	-	-	7.95 [4.20, 15.97]

Table 9: Average MIA ROC AUC Across Models (Mean [Min, Max], N=5)

Method	$\varepsilon = 0.1$	$\varepsilon = 0.5$	$\varepsilon = 1.0$	$\varepsilon = 2.0$	$\varepsilon = \infty$
DPSGD (1e)	0.511 [0.506, 0.516]	0.512 [0.507, 0.520]	0.512 [0.507, 0.522]	0.513 [0.508, 0.524]	-
DPSGD (3e)	0.511 [0.509, 0.513]	0.512 [0.509, 0.517]	0.513 [0.509, 0.521]	0.514 [0.509, 0.524]	-
DPSGD (5e)	0.509 [0.507, 0.510]	0.509 [0.505, 0.513]	0.509 [0.505, 0.513]	0.510 [0.505, 0.513]	-
FERRET-MAX (1e)	0.509 [0.506, 0.512]	0.508 [0.506, 0.510]	0.507 [0.505, 0.510]	0.507 [0.505, 0.508]	-
FERRET-MAX (3e)	0.509 [0.506, 0.510]	0.512 [0.507, 0.519]	0.512 [0.507, 0.518]	0.511 [0.508, 0.514]	-
FERRET-MAX (5e)	0.510 [0.508, 0.511]	0.513 [0.508, 0.526]	0.512 [0.508, 0.523]	0.512 [0.508, 0.521]	-
FERRET-EIGHTH (1e)	0.514 [0.508, 0.524]	0.512 [0.508, 0.519]	0.509 [0.505, 0.513]	0.507 [0.505, 0.510]	-
FERRET-EIGHTH (3e)	0.516 [0.506, 0.528]	0.522 [0.506, 0.557]	0.522 [0.506, 0.560]	0.516 [0.508, 0.537]	-
FERRET-EIGHTH (5e)	0.518 [0.507, 0.540]	0.533 [0.506, 0.591]	0.534 [0.506, 0.600]	0.529 [0.506, 0.577]	-
FERRET-2 (1e)	0.521 [0.511, 0.538]	0.524 [0.515, 0.541]	0.511 [0.505, 0.515]	0.507 [0.502, 0.510]	-
FERRET-2 (3e)	0.535 [0.523, 0.551]	0.571 [0.545, 0.618]	0.572 [0.537, 0.630]	0.546 [0.530, 0.574]	-
FERRET-2 (5e)	0.541 [0.530, 0.557]	0.549 [0.525, 0.578]	0.548 [0.524, 0.574]	0.546 [0.524, 0.571]	-
Non-DP (1e)	-	-	-	-	0.759 [0.569, 0.912]
Non-DP (3e)	-	-	-	-	0.974 [0.893, 0.999]
Non-DP (5e)	-	-	-	-	0.995 [0.976, 1.000]

Table 10: Average MIA Advantage Across Models (Mean [Min, Max], N=5)

Method	$\varepsilon = 0.1$	$\varepsilon = 0.5$	$\varepsilon = 1.0$	$\varepsilon = 2.0$	$\varepsilon = \infty$
DPSGD (1e)	0.022 [0.016, 0.033]	0.022 [0.015, 0.036]	0.023 [0.015, 0.040]	0.024 [0.016, 0.044]	-
DPSGD (3e)	0.021 [0.017, 0.027]	0.023 [0.019, 0.032]	0.024 [0.019, 0.037]	0.025 [0.017, 0.043]	-
DPSGD (5e)	0.020 [0.017, 0.024]	0.019 [0.012, 0.024]	0.019 [0.013, 0.024]	0.020 [0.014, 0.024]	-
FERRET-MAX (1e)	0.020 [0.016, 0.029]	0.017 [0.016, 0.019]	0.016 [0.014, 0.020]	0.015 [0.014, 0.016]	-
FERRET-MAX (3e)	0.018 [0.015, 0.022]	0.024 [0.017, 0.036]	0.022 [0.013, 0.031]	0.019 [0.015, 0.022]	-
FERRET-MAX (5e)	0.020 [0.018, 0.023]	0.025 [0.019, 0.045]	0.023 [0.016, 0.039]	0.022 [0.016, 0.035]	-
FERRET-EIGHTH (1e)	0.025 [0.018, 0.040]	0.022 [0.018, 0.032]	0.019 [0.017, 0.020]	0.017 [0.015, 0.019]	-
FERRET-EIGHTH (3e)	0.028 [0.016, 0.046]	0.038 [0.016, 0.089]	0.039 [0.016, 0.097]	0.030 [0.017, 0.060]	-
FERRET-EIGHTH (5e)	0.032 [0.018, 0.066]	0.054 [0.016, 0.138]	0.056 [0.017, 0.155]	0.048 [0.016, 0.122]	-
FERRET-2 (1e)	0.039 [0.024, 0.071]	0.040 [0.026, 0.066]	0.020 [0.012, 0.027]	0.016 [0.013, 0.018]	-
FERRET-2 (3e)	0.057 [0.042, 0.083]	0.117 [0.069, 0.199]	0.117 [0.059, 0.211]	0.075 [0.050, 0.115]	-
FERRET-2 (5e)	0.067 [0.048, 0.096]	0.079 [0.042, 0.137]	0.076 [0.039, 0.121]	0.071 [0.039, 0.110]	-
Non-DP (1e)	-	-	-	-	0.429 [0.105, 0.700]
Non-DP (3e)	-	-	-	-	0.882 [0.644, 0.982]
Non-DP (5e)	-	-	-	-	0.960 [0.867, 0.994]

Table 11: Average Training Time (s) Across Models (Mean [Min, Max], N=5)

Method	$\varepsilon = 0.1$	$\varepsilon = 0.5$	$\varepsilon = 1.0$	$\varepsilon = 2.0$	$\varepsilon = \infty$
DPSGD (1e)	1540 [320, 3470]	1540 [320, 3480]	1540 [320, 3460]	1540 [330, 3470]	-
DPSGD (3e)	4590 [970, 10340]	4590 [970, 10330]	4590 [970, 10360]	4600 [970, 10400]	-
DPSGD (5e)	7780 [1610, 17100]	7790 [1610, 17120]	7810 [1610, 17170]	7840 [1610, 17380]	-
FERRET-MAX (1e)	330 [70, 740]	410 [80, 880]	470 [90, 1050]	540 [110, 1200]	-
FERRET-MAX (3e)	950 [180, 2130]	1040 [200, 2280]	1130 [230, 2510]	1290 [260, 2830]	-
FERRET-MAX (5e)	1500 [290, 3330]	1600 [320, 3570]	1720 [340, 3800]	1900 [380, 4190]	-
FERRET-EIGHTH (1e)	350 [70, 780]	470 [90, 1050]	560 [110, 1240]	620 [120, 1380]	-
FERRET-EIGHTH (3e)	950 [180, 2100]	1090 [210, 2420]	1240 [240, 2740]	1490 [280, 3300]	-
FERRET-EIGHTH (5e)	1500 [300, 3290]	1670 [320, 3690]	1840 [350, 4080]	2130 [410, 4760]	-
FERRET-2 (1e)	370 [70, 810]	580 [110, 1300]	770 [140, 1730]	930 [160, 2240]	-
FERRET-2 (3e)	960 [190, 2120]	1200 [230, 2650]	1480 [280, 3300]	1930 [370, 4310]	-
FERRET-2 (5e)	1540 [300, 3410]	1790 [350, 3980]	2080 [400, 4630]	2620 [490, 5860]	-
Non-DP (1e)	-	-	-	-	910 [150, 2210]
Non-DP (3e)	-	-	-	-	2690 [460, 6560]
Non-DP (5e)	-	-	-	-	4440 [760, 10830]

10.1 Comprehensive Comparison of Methods, Models, and Privacy Budgets

Table 12: Comprehensive Comparison of Methods, Models, and Privacy Budgets

Method	Model	Test PPL					MIA AUC					Train Time (s)				
		$\varepsilon = 0.1$	$\varepsilon = 0.5$	$\varepsilon = 1.0$	$\varepsilon = 2.0$	$\varepsilon = \infty$	$\varepsilon = 0.1$	$\varepsilon = 0.5$	$\varepsilon = 1.0$	$\varepsilon = 2.0$	$\varepsilon = \infty$	$\varepsilon = 0.1$	$\varepsilon = 0.5$	$\varepsilon = 1.0$	$\varepsilon = 2.0$	$\varepsilon = \infty$
DPSGD (1e)	DeepSeek-1.5B	4.79	4.24	4.06	3.89	-	0.510	0.510	0.509	0.509	-	3473	3484	3458	3469	-
	SmolLM-360M	4.53	4.13	3.99	3.86	-	0.506	0.507	0.507	0.508	-	860	860	862	861	-
	TinyLlama-1.1B	3.65	3.16	2.97	2.82	-	0.509	0.509	0.509	0.509	-	1846	1844	1844	1846	-
	BLOOM-560M	22.39	13.74	11.75	10.36	-	0.516	0.520	0.522	0.524	-	1199	1198	1198	1198	-
	GPT-2	10.44	9.41	9.09	8.79	-	0.513	0.513	0.513	0.514	-	325	324	325	325	-
	Mean	**9.16**	**6.94**	**6.37**	**5.94**	-	**0.511**	**0.512**	**0.512**	**0.513**	-	**1541**	**1542**	**1537**	**1540**	-
DPSGD (3e)	DeepSeek-1.5B	4.84	3.83	3.61	3.47	-	0.509	0.509	0.509	0.509	-	10343	10329	10356	10399	-
	SmolLM-360M	4.26	3.60	3.44	3.31	-	0.509	0.511	0.511	0.511	-	2561	2560	2560	2562	-
	TinyLlama-1.1B	5.36	3.71	3.47	3.30	-	0.513	0.509	0.509	0.509	-	5495	5494	5490	5499	-
	BLOOM-560M	136.94	25.98	19.70	16.10	-	0.511	0.517	0.521	0.524	-	3574	3580	3569	3574	-
	GPT-2	10.45	8.42	7.97	7.59	-	0.512	0.514	0.514	0.515	-	968	967	969	969	-
	Mean	**32.37**	**9.11**	**7.64**	**6.76**	-	**0.511**	**0.512**	**0.513**	**0.514**	-	**4588**	**4586**	**4589**	**4601**	-
DPSGD (5e)	DeepSeek-1.5B	5.10	3.86	3.66	3.54	-	0.510	0.509	0.510	0.511	-	17098	17125	17173	17379	-
	SmolLM-360M	4.14	3.38	3.20	3.06	-	0.510	0.511	0.511	0.510	-	4283	4285	4284	4288	-
	TinyLlama-1.1B	10.16	5.08	4.53	4.25	-	0.509	0.509	0.510	0.510	-	10002	10004	10047	10000	-
	BLOOM-560M	908.53	37.76	30.83	25.72	-	0.507	0.505	0.505	0.505	-	5931	5927	5936	5939	-
	GPT-2	10.68	7.99	7.44	7.02	-	0.508	0.513	0.513	0.513	-	1608	1606	1608	1609	-
	Mean	**187.72**	**11.61**	**9.93**	**8.72**	-	**0.509**	**0.509**	**0.509**	**0.510**	-	**7785**	**7789**	**7810**	**7843**	-
FERRET-MAX (1e)	DeepSeek-1.5B	4.69	5.48	5.72	6.91	-	0.509	0.507	0.506	0.508	-	736	884	1046	1198	-
	SmolLM-360M	4.73	4.88	5.28	5.38	-	0.506	0.506	0.506	0.506	-	176	220	251	296	-
	TinyLlama-1.1B	2.43	2.49	2.53	3.83	-	0.509	0.510	0.510	0.507	-	425	521	593	664	-
	BLOOM-560M	6.53	6.68	7.47	43.22	-	0.512	0.509	0.510	0.508	-	259	324	374	426	-
	GPT-2	9.43	10.81	11.83	13.57	-	0.510	0.509	0.505	0.505	-	67	81	94	107	-
	Mean	**5.57**	**6.07**	**6.56**	**14.58**	-	**0.509**	**0.508**	**0.507**	**0.507**	-	**332**	**406**	**472**	**538**	-
FERRET-MAX (3e)	DeepSeek-1.5B	3.40	3.49	3.46	4.41	-	0.507	0.507	0.507	0.508	-	2131	2281	2507	2834	-
	SmolLM-360M	4.62	4.11	4.35	4.71	-	0.506	0.508	0.508	0.508	-	505	561	613	717	-
	TinyLlama-1.1B	2.39	2.38	2.42	2.39	-	0.510	0.519	0.518	0.513	-	1191	1320	1435	1618	-
	BLOOM-560M	6.22	9.51	6.71	7.22	-	0.510	0.515	0.514	0.514	-	725	813	887	1015	-
	GPT-2	8.63	7.90	8.53	9.25	-	0.510	0.513	0.511	0.511	-	185	204	225	256	-
	Mean	**5.05**	**5.48**	**5.09**	**5.59**	-	**0.509**	**0.512**	**0.512**	**0.511**	-	**947**	**1036**	**1134**	**1288**	-
FERRET-MAX (5e)	DeepSeek-1.5B	3.21	3.22	3.28	3.53	-	0.510	0.511	0.508	0.509	-	3327	3573	3802	4186	-
	SmolLM-360M	4.46	3.78	3.76	4.13	-	0.508	0.508	0.508	0.508	-	802	864	927	1018	-
	TinyLlama-1.1B	2.44	2.40	2.38	2.39	-	0.510	0.526	0.523	0.521	-	1901	2045	2172	2462	-
	BLOOM-560M	9.82	4.81	6.12	6.22	-	0.510	0.510	0.511	0.510	-	1151	1222	1374	1463	-
	GPT-2	9.12	7.17	7.32	7.93	-	0.511	0.511	0.512	0.511	-	295	317	341	378	-
	Mean	**5.81**	**4.28**	**4.57**	**4.84**	-	**0.510**	**0.513**	**0.512**	**0.512**	-	**1495**	**1604**	**1723**	**1902**	-
FERRET-EIGHTH (1e)	DeepSeek-1.5B	3.30	3.82	4.69	5.27	-	0.509	0.509	0.509	0.508	-	779	1053	1236	1381	-
	SmolLM-360M	3.38	3.93	4.72	4.91	-	0.508	0.508	0.505	0.505	-	187	252	305	344	-
	TinyLlama-1.1B	2.48	2.52	2.60	3.12	-	0.524	0.519	0.513	0.510	-	455	606	712	796	-
	BLOOM-560M	10.85	11.46	18.53	16.46	-	0.515	0.514	0.510	0.507	-	264	350	416	469	-
	GPT-2	7.21	7.39	8.35	12.22	-	0.511	0.508	0.510	0.505	-	70	92	107	121	-
	Mean	**5.45**	**5.83**	**7.78**	**8.40**	-	**0.514**	**0.512**	**0.509**	**0.507**	-	**351**	**471**	**555**	**622**	-
FERRET-EIGHTH (3e)	DeepSeek-1.5B	3.19	3.19	3.23	3.34	-	0.517	0.512	0.509	0.508	-	2099	2419	2744	3302	-
	SmolLM-360M	3.29	2.86	3.29	3.71	-	0.506	0.506	0.506	0.508	-	500	586	689	822	-
	TinyLlama-1.1B	2.46	2.56	2.53	2.57	-	0.528	0.557	0.560	0.537	-	1230	1398	1599	1903	-
	BLOOM-560M	6.15	6.12	6.48	6.54	-	0.513	0.525	0.523	0.517	-	724	840	945	1141	-
	GPT-2	6.62	6.05	5.96	6.39	-	0.513	0.512	0.512	0.512	-	185	210	242	283	-
	Mean	**4.34**	**4.16**	**4.30**	**4.51**	-	**0.516**	**0.522**	**0.522**	**0.516**	-	**947**	**1091**	**1244**	**1490**	-
FERRET-EIGHTH (5e)	DeepSeek-1.5B	3.18	3.21	3.16	3.20	-	0.517	0.523	0.518	0.513	-	3288	3685	4081	4760	-
	SmolLM-360M	3.06	2.75	2.80	3.04	-	0.507	0.506	0.506	0.506	-	803	894	990	1152	-
	TinyLlama-1.1B	2.51	2.56	2.57	2.58	-	0.540	0.591	0.600	0.577	-	1960	2149	2384	2715	-
	BLOOM-560M	6.56	5.92	5.93	5.98	-	0.516	0.531	0.533	0.533	-	1144	1276	1405	1633	-
	GPT-2	6.46	5.46	5.56	5.47	-	0.512	0.514	0.514	0.515	-	295	323	355	415	-
	Mean	**4.35**	**3.98**	**4.00**	**4.05**	-	**0.518**	**0.533**	**0.534**	**0.529**	-	**1498**	**1665**	**1843**	**2135**	-
FERRET-2 (1e)	DeepSeek-1.5B	4.35	3.91	5.08	6.98	-	0.518	0.521	0.512	0.509	-	814	1297	1726	2244	-
	SmolLM-360M	2.60	2.57	3.10	5.19	-	0.521	0.520	0.513	0.506	-	198	318	418	465	-
	TinyLlama-1.1B	3.96	3.79	428515.02	5.00	-	0.538	0.541	0.505	0.508	-	493	750	991	1134	-
	BLOOM-560M	226.95	14.72	18719.68	27019.12	-	0.511	0.520	0.508	0.502	-	271	427	574	644	-
	GPT-2	5.21	5.55	6.72	15.14	-	0.518	0.515	0.515	0.510	-	74	108	144	161	-
	Mean	**48.61**	**6.11**	**89449.92**	**5410.29**	-	**0.521**	**0.524**	**0.511**	**0.507**	-	**370**	**580**	**771**	**929**	-
FERRET-2 (3e)	DeepSeek-1.5B	4.62	4.37	4.32	4.51	-	0.533	0.551	0.540	0.530	-	2124	2654	3303	4314	-
	SmolLM-360M	2.54	2.49	2.51	2.53	-	0.533	0.564	0.560	0.541	-	512	652	788	1025	-
	TinyLlama-1.1B	3.91	4.02	4.27	4.12	-	0.551	0.618	0.630	0.574	-	1259	1550	1931	2511	-
	BLOOM-560M	15.90	15.27	9.87	16.32	-	0.534	0.578	0.592	0.555	-	731	913	1115	1439	-
	GPT-2	5.26	4.85	4.84	5.09	-	0.523	0.545	0.537	0.530	-	187	228	278	369	-
	Mean	**6.45**	**6.20**	**5.16**	**6.52**	-	**0.535**	**0.571**	**0.572**	**0.546**	-	**963**	**1200**	**1483**	**1932**	-
FERRET-2 (5e)	DeepSeek-1.5B	5.11	4.67	4.43	4.63	-	0.548	0.578	0.566	0.566	-	3410	3981	4625	5864	-
	SmolLM-360M	2.54	2.49	2.49	2.49	-	0.531	0.530	0.529	0.529	-	825	960	1125	1414	-
	TinyLlama-1.1B	3.87	4.12	4.58	4.07	-	0.557	0.574	0.574	0.571	-	1995	2312	2676	3357	-
	BLOOM-560M	12.66	13.74	13.41	13.53	-	0.536	0.537	0.546	0.537	-	1172	1354	1579	1958	-
	GPT-2	5.21	4.82	4.73	4.74	-	0.530	0.525	0.524	0.524	-	302	349	403	495	-
	Mean	**5.88**	**5.97**	**5.93**	**5.89**	-	**0.541**	**0.549**	**0.548**	**0.546**	-	**1541**	**1791**	**2082**	**2617**	-
Non-DP (1e)	DeepSeek-1.5B	-	-	-	-	2.83	-	-	-	-	0.912	-	-	-	-	2212
	SmolLM-360M	-	-	-	-	2.42	-	-	-	-	0.612	-	-	-	-	449
	TinyLlama-1.1B	-	-	-	-	2.39	-	-	-	-	0.899	-	-	-	-	1091
	BLOOM-560M	-	-	-	-	4.16	-	-	-	-	0.804	-	-	-	-	631
	GPT-2	-	-	-	-	4.42	-	-	-	-	0.569	-	-	-	-	155
	Mean	-	-	-	-	**3.25**	-	-	-	-	**0.759**	-	-	-	-	**908**
Non-DP (3e)	DeepSeek-1.5B	-	-	-	-	5.98	-	-	-	-	0.999	-	-	-	-	6555
	SmolLM-360M	-	-	-	-	3.23	-	-	-	-	0.993	-	-	-	-	1338
	TinyLlama-1.1B	-	-	-	-	5.77	-	-	-	-	0.993	-	-	-	-	3227
	BLOOM-560M	-	-	-	-	10.36	-	-	-	-	0.991	-	-	-	-	1865
	GPT-2	-	-	-	-	4.50	-	-	-	-	0.893	-	-	-	-	460
	Mean	-	-	-	-	**5.97**	-	-	-	-	**0.974**	-	-	-	-	**2689**
Non-DP (5e)	DeepSeek-1.5B	-	-	-	-	7.54	-	-	-	-	1.000	-	-	-	-	10828
	SmolLM-360M	-	-	-	-	4.20	-	-	-	-	1.000	-	-	-	-	2209
	TinyLlama-1.1B	-	-	-	-	7.05	-	-	-	-	0.999	-	-	-	-	5333
	BLOOM-560M	-	-	-	-	15.97	-	-	-	-	0.998	-	-	-	-	3091
	GPT-2	-	-	-	-	5.00	-	-	-	-	0.976	-	-	-	-	761
	Mean	-	-	-	-	**7.95**	-	-	-	-	**0.995**	-	-	-	-	**4444**

Table 13: Comparison of Privacy Metrics Across Methods, Models, and Privacy Budgets

		$\varepsilon = 0.1$		$\varepsilon = 0.5$		$\varepsilon = 1.0$		$\varepsilon = 2.0$		$\varepsilon = \infty$	
Method	**Model**	**ROC AUC**	**Advantage**	**ROC AUC**	**Advantage**	**ROC AUC**	**Advantage**	**ROC AUC**	**Advantage**	**ROC AUC**	**Advantage**
	DeepSeek-1.5B	0.510	0.018	0.510	0.018	0.509	0.016	0.509	0.016	-	-
	SmolLM-360M	0.506	0.016	0.507	0.015	0.507	0.015	0.508	0.016	-	-
DPSGD (1e)	TinyLlama-1.1B	0.509	0.016	0.509	0.019	0.509	0.019	0.509	0.018	-	-
	BLOOM-560M	0.516	0.033	0.520	0.036	0.522	0.040	0.524	0.044	-	-
	GPT-2	0.513	0.028	0.513	0.022	0.513	0.024	0.514	0.028	-	-
	Mean	**0.511**	**0.022**	**0.512**	**0.022**	**0.512**	**0.023**	**0.513**	**0.024**	-	-
	DeepSeek-1.5B	0.509	0.017	0.509	0.019	0.509	0.020	0.509	0.017	-	-
	SmolLM-360M	0.509	0.018	0.511	0.019	0.511	0.020	0.511	0.019	-	-
DPSGD (3e)	TinyLlama-1.1B	0.513	0.027	0.509	0.021	0.509	0.019	0.509	0.019	-	-
	BLOOM-560M	0.511	0.018	0.517	0.032	0.521	0.037	0.524	0.043	-	-
	GPT-2	0.512	0.026	0.514	0.024	0.514	0.023	0.515	0.024	-	-
	Mean	**0.511**	**0.021**	**0.512**	**0.023**	**0.513**	**0.024**	**0.514**	**0.025**	-	-
	DeepSeek-1.5B	0.510	0.017	0.509	0.016	0.510	0.018	0.511	0.019	-	-
	SmolLM-360M	0.510	0.023	0.511	0.024	0.511	0.023	0.510	0.022	-	-
DPSGD (5e)	TinyLlama-1.1B	0.509	0.024	0.509	0.020	0.510	0.018	0.510	0.023	-	-
	BLOOM-560M	0.507	0.017	0.505	0.012	0.505	0.013	0.505	0.014	-	-
	GPT-2	0.508	0.018	0.513	0.024	0.513	0.024	0.513	0.024	-	-
	Mean	**0.509**	**0.020**	**0.509**	**0.019**	**0.509**	**0.019**	**0.510**	**0.020**	-	-
	DeepSeek-1.5B	0.509	0.018	0.507	0.017	0.506	0.014	0.508	0.014	-	-
	SmolLM-360M	0.506	0.016	0.506	0.016	0.506	0.015	0.506	0.014	-	-
FERRET-MAX (1e)	TinyLlama-1.1B	0.509	0.017	0.510	0.019	0.510	0.018	0.507	0.016	-	-
	BLOOM-560M	0.512	0.029	0.509	0.016	0.510	0.020	0.508	0.016	-	-
	GPT-2	0.510	0.019	0.509	0.017	0.505	0.016	0.505	0.014	-	-
	Mean	**0.509**	**0.020**	**0.508**	**0.017**	**0.507**	**0.016**	**0.507**	**0.015**	-	-
	DeepSeek-1.5B	0.507	0.015	0.507	0.018	0.507	0.013	0.508	0.015	-	-
	SmolLM-360M	0.506	0.016	0.508	0.017	0.508	0.019	0.508	0.018	-	-
FERRET-MAX (3e)	TinyLlama-1.1B	0.510	0.020	0.519	0.036	0.518	0.031	0.513	0.020	-	-
	BLOOM-560M	0.510	0.022	0.515	0.026	0.514	0.025	0.514	0.022	-	-
	GPT-2	0.510	0.017	0.513	0.024	0.511	0.020	0.511	0.021	-	-
	Mean	**0.509**	**0.018**	**0.512**	**0.024**	**0.512**	**0.022**	**0.511**	**0.019**	-	-
	DeepSeek-1.5B	0.510	0.020	0.511	0.021	0.508	0.016	0.509	0.016	-	-
	SmolLM-360M	0.508	0.018	0.508	0.019	0.508	0.018	0.508	0.017	-	-
FERRET-MAX (5e)	TinyLlama-1.1B	0.510	0.019	0.526	0.045	0.523	0.039	0.521	0.035	-	-
	BLOOM-560M	0.510	0.020	0.510	0.019	0.511	0.020	0.510	0.020	-	-
	GPT-2	0.511	0.023	0.511	0.022	0.512	0.021	0.511	0.023	-	-
	Mean	**0.510**	**0.020**	**0.513**	**0.025**	**0.512**	**0.023**	**0.512**	**0.022**	-	-
	DeepSeek-1.5B	0.509	0.018	0.509	0.018	0.509	0.017	0.508	0.017	-	-
	SmolLM-360M	0.508	0.018	0.508	0.019	0.505	0.017	0.505	0.016	-	-
FERRET-EIGHTH (1e)	TinyLlama-1.1B	0.524	0.040	0.519	0.032	0.513	0.020	0.510	0.019	-	-
	BLOOM-560M	0.515	0.026	0.514	0.022	0.510	0.020	0.507	0.016	-	-
	GPT-2	0.511	0.022	0.508	0.018	0.510	0.019	0.505	0.015	-	-
	Mean	**0.514**	**0.025**	**0.512**	**0.022**	**0.509**	**0.019**	**0.507**	**0.017**	-	-
	DeepSeek-1.5B	0.517	0.031	0.512	0.023	0.509	0.021	0.508	0.017	-	-
	SmolLM-360M	0.506	0.016	0.506	0.016	0.506	0.016	0.508	0.017	-	-
FERRET-EIGHTH (3e)	TinyLlama-1.1B	0.528	0.046	0.557	0.089	0.560	0.097	0.537	0.060	-	-
	BLOOM-560M	0.513	0.023	0.525	0.040	0.523	0.039	0.517	0.031	-	-
	GPT-2	0.513	0.023	0.512	0.022	0.512	0.024	0.512	0.022	-	-
	Mean	**0.516**	**0.028**	**0.522**	**0.038**	**0.522**	**0.039**	**0.516**	**0.030**	-	-
	DeepSeek-1.5B	0.517	0.030	0.523	0.043	0.518	0.033	0.513	0.023	-	-
	SmolLM-360M	0.507	0.018	0.506	0.016	0.506	0.017	0.506	0.016	-	-
FERRET-EIGHTH (5e)	TinyLlama-1.1B	0.540	0.066	0.591	0.138	0.600	0.155	0.577	0.122	-	-
	BLOOM-560M	0.516	0.026	0.531	0.047	0.533	0.053	0.533	0.050	-	-
	GPT-2	0.512	0.021	0.514	0.024	0.514	0.023	0.515	0.026	-	-
	Mean	**0.518**	**0.032**	**0.533**	**0.054**	**0.534**	**0.056**	**0.529**	**0.048**	-	-
	DeepSeek-1.5B	0.518	0.031	0.521	0.039	0.512	0.020	0.509	0.015	-	-
	SmolLM-360M	0.521	0.036	0.520	0.035	0.513	0.021	0.506	0.015	-	-
FERRET-2 (1e)	TinyLlama-1.1B	0.538	0.071	0.541	0.066	0.505	0.012	0.508	0.018	-	-
	BLOOM-560M	0.511	0.024	0.520	0.034	0.508	0.018	0.502	0.013	-	-
	GPT-2	0.518	0.033	0.515	0.026	0.515	0.027	0.510	0.018	-	-
	Mean	**0.521**	**0.039**	**0.524**	**0.040**	**0.511**	**0.020**	**0.507**	**0.016**	-	-
	DeepSeek-1.5B	0.533	0.059	0.551	0.083	0.540	0.079	0.530	0.050	-	-
	SmolLM-360M	0.533	0.050	0.564	0.103	0.560	0.092	0.541	0.063	-	-
FERRET-2 (3e)	TinyLlama-1.1B	0.551	0.083	0.618	0.199	0.630	0.211	0.574	0.115	-	-
	BLOOM-560M	0.534	0.054	0.578	0.130	0.592	0.145	0.555	0.096	-	-
	GPT-2	0.523	0.042	0.545	0.069	0.537	0.059	0.530	0.050	-	-
	Mean	**0.535**	**0.057**	**0.571**	**0.117**	**0.572**	**0.117**	**0.546**	**0.075**	-	-
	DeepSeek-1.5B	0.548	0.080	0.578	0.137	0.566	0.121	0.566	0.110	-	-
	SmolLM-360M	0.531	0.048	0.530	0.049	0.529	0.047	0.529	0.048	-	-
FERRET-2 (5e)	TinyLlama-1.1B	0.557	0.096	0.574	0.109	0.574	0.108	0.571	0.100	-	-
	BLOOM-560M	0.536	0.059	0.537	0.057	0.546	0.067	0.537	0.057	-	-
	GPT-2	0.530	0.052	0.525	0.042	0.524	0.039	0.524	0.039	-	-
	Mean	**0.541**	**0.067**	**0.549**	**0.079**	**0.548**	**0.076**	**0.546**	**0.071**	-	-
	DeepSeek-1.5B	-	-	-	-	-	-	-	-	0.912	0.700
	SmolLM-360M	-	-	-	-	-	-	-	-	0.612	0.177
Non-DP (1e)	TinyLlama-1.1B	-	-	-	-	-	-	-	-	0.899	0.681
	BLOOM-560M	-	-	-	-	-	-	-	-	0.804	0.484
	GPT-2	-	-	-	-	-	-	-	-	0.569	0.105
	Mean	-	-	-	-	-	-	-	-	**0.759**	**0.429**
	DeepSeek-1.5B	-	-	-	-	-	-	-	-	0.999	0.982
	SmolLM-360M	-	-	-	-	-	-	-	-	0.993	0.934
Non-DP (3e)	TinyLlama-1.1B	-	-	-	-	-	-	-	-	0.993	0.931
	BLOOM-560M	-	-	-	-	-	-	-	-	0.991	0.920
	GPT-2	-	-	-	-	-	-	-	-	0.893	0.644
	Mean	-	-	-	-	-	-	-	-	**0.974**	**0.882**
	DeepSeek-1.5B	-	-	-	-	-	-	-	-	1.000	0.994
	SmolLM-360M	-	-	-	-	-	-	-	-	1.000	0.990
Non-DP (5e)	TinyLlama-1.1B	-	-	-	-	-	-	-	-	0.999	0.975
	BLOOM-560M	-	-	-	-	-	-	-	-	0.998	0.971
	GPT-2	-	-	-	-	-	-	-	-	0.976	0.867
	Mean	-	-	-	-	-	-	-	-	**0.995**	**0.960**

Table 14: Comparison of Utility Metrics Across Methods, Models, and Privacy Budgets

		$\varepsilon = 0.1$			$\varepsilon = 0.5$			$\varepsilon = 1.0$			$\varepsilon = 2.0$			$\varepsilon = \infty$			$\varepsilon = 0.1$	$\varepsilon = 0.5$	$\varepsilon = 1.0$	$\varepsilon = 2.0$	$\varepsilon = \infty$
Method	**Model**	**Train**	**Test**	**Gap**	**Train**	**Test**	**Gap**	**Train**	**Test**	**Gap**	**Train**	**Test**	**Gap**	**Train**	**Test**	**Gap**	**Time (s)**	**Time (s)**	**Time (s)**	**Time (s)**	**Time (s)**
DPSGD (1e)	DeepSeek-1.5B	4.74	4.79	0.05	4.20	4.24	0.04	4.02	4.06	0.04	3.86	3.89	0.04	-	-	-	3473	3484	3458	3469	-
	SmolLM-360M	4.51	4.53	0.02	4.11	4.13	0.02	3.97	3.99	0.02	3.84	3.86	0.02	-	-	-	860	860	862	861	-
	TinyLlama-1.1B	3.64	3.65	0.01	3.15	3.16	0.01	2.96	2.97	0.01	2.80	2.82	0.01	-	-	-	1846	1844	1844	1846	-
	BLOOM-560M	21.91	22.39	0.48	13.37	13.74	0.36	11.41	11.75	0.34	10.03	10.36	0.32	-	-	-	1199	1198	1198	1198	-
	GPT-2	10.42	10.44	0.02	9.38	9.41	0.03	9.05	9.09	0.03	8.75	8.79	0.04	-	-	-	325	324	325	325	-
	Mean	**9.05**	**9.16**	**0.11**	**6.84**	**6.94**	**0.09**	**6.28**	**6.37**	**0.09**	**5.86**	**5.94**	**0.09**	-	-	-	**1541**	**1542**	**1537**	**1540**	-
DPSGD (3e)	DeepSeek-1.5B	4.79	4.84	0.05	3.79	3.83	0.04	3.58	3.61	0.04	3.43	3.47	0.04	-	-	-	10343	10329	10356	10399	-
	SmolLM-360M	4.24	4.26	0.02	3.58	3.60	0.02	3.43	3.44	0.02	3.30	3.31	0.02	-	-	-	2561	2560	2560	2562	-
	TinyLlama-1.1B	5.33	5.36	0.03	3.69	3.71	0.02	3.45	3.47	0.02	3.28	3.30	0.02	-	-	-	5495	5494	5490	5499	-
	BLOOM-560M	133.61	136.94	3.33	24.93	25.98	1.05	18.80	19.70	0.90	15.30	16.10	0.81	-	-	-	3574	3580	3569	3574	-
	GPT-2	10.41	10.45	0.03	8.37	8.42	0.05	7.92	7.97	0.05	7.54	7.59	0.05	-	-	-	968	967	969	969	-
	Mean	**31.68**	**32.37**	**0.69**	**8.87**	**9.11**	**0.24**	**7.43**	**7.64**	**0.21**	**6.57**	**6.76**	**0.19**	-	-	-	**4588**	**4586**	**4589**	**4601**	-
DPSGD (5e)	DeepSeek-1.5B	5.04	5.10	0.05	3.82	3.86	0.04	3.62	3.66	0.04	3.50	3.54	0.04	-	-	-	17098	17125	17173	17379	-
	SmolLM-360M	4.12	4.14	0.02	3.36	3.38	0.02	3.18	3.20	0.02	3.05	3.06	0.02	-	-	-	4283	4285	4284	4288	-
	TinyLlama-1.1B	10.09	10.16	0.07	5.05	5.08	0.03	4.50	4.53	0.03	4.21	4.25	0.03	-	-	-	10002	10004	10047	10000	-
	BLOOM-560M	896.24	908.53	12.29	36.62	37.76	1.14	29.55	30.83	1.28	24.45	25.72	1.27	-	-	-	5931	5927	5936	5939	-
	GPT-2	10.65	10.68	0.02	7.94	7.99	0.05	7.38	7.44	0.05	6.97	7.02	0.06	-	-	-	1608	1606	1608	1609	-
	Mean	**185.23**	**187.72**	**2.49**	**11.36**	**11.61**	**0.26**	**9.65**	**9.93**	**0.28**	**8.43**	**8.72**	**0.29**	-	-	-	**7785**	**7789**	**7810**	**7843**	-
FERRET-MAX (1e)	DeepSeek-1.5B	4.66	4.69	0.04	5.44	5.48	0.04	5.67	5.72	0.04	6.86	6.91	0.05	-	-	-	736	884	1046	1198	-
	SmolLM-360M	4.72	4.73	0.01	4.86	4.88	0.01	5.26	5.28	0.01	5.37	5.38	0.01	-	-	-	176	220	251	296	-
	TinyLlama-1.1B	2.42	2.43	0.01	2.48	2.49	0.01	2.52	2.53	0.01	3.83	3.83	0.00	-	-	-	425	521	593	664	-
	BLOOM-560M	6.48	6.53	0.05	6.64	6.68	0.04	7.43	7.47	0.04	42.76	43.22	0.46	-	-	-	259	324	374	426	-
	GPT-2	9.43	9.43	0.01	10.81	10.81	-0.00	11.85	11.83	-0.02	13.61	13.57	-0.03	-	-	-	67	81	94	107	-
	Mean	**5.54**	**5.57**	**0.02**	**6.05**	**6.07**	**0.02**	**6.55**	**6.56**	**0.02**	**14.48**	**14.58**	**0.10**	-	-	-	**332**	**406**	**472**	**538**	-
FERRET-MAX (3e)	DeepSeek-1.5B	3.37	3.40	0.03	3.46	3.49	0.03	3.43	3.46	0.03	4.37	4.41	0.04	-	-	-	2131	2281	2507	2834	-
	SmolLM-360M	4.61	4.62	0.02	4.09	4.11	0.02	4.33	4.35	0.02	4.69	4.71	0.01	-	-	-	505	561	613	717	-
	TinyLlama-1.1B	2.38	2.39	0.02	2.36	2.38	0.03	2.40	2.42	0.03	2.37	2.39	0.02	-	-	-	1191	1320	1435	1618	-
	BLOOM-560M	6.16	6.22	0.06	9.31	9.51	0.19	6.60	6.71	0.10	7.13	7.22	0.09	-	-	-	725	813	887	1015	-
	GPT-2	8.61	8.63	0.02	7.88	7.90	0.03	8.51	8.53	0.02	9.23	9.25	0.02	-	-	-	185	204	225	256	-
	Mean	**5.03**	**5.05**	**0.03**	**5.42**	**5.48**	**0.06**	**5.06**	**5.09**	**0.04**	**5.56**	**5.59**	**0.04**	-	-	-	**947**	**1036**	**1134**	**1288**	-
FERRET-MAX (5e)	DeepSeek-1.5B	3.18	3.21	0.03	3.18	3.22	0.03	3.24	3.28	0.03	3.50	3.53	0.03	-	-	-	3327	3573	3802	4186	-
	SmolLM-360M	4.45	4.46	0.02	3.76	3.78	0.02	3.74	3.76	0.02	4.12	4.13	0.02	-	-	-	802	864	927	1018	-
	TinyLlama-1.1B	2.43	2.44	0.02	2.36	2.40	0.04	2.35	2.38	0.03	2.36	2.39	0.03	-	-	-	1901	2045	2172	2462	-
	BLOOM-560M	9.67	9.82	0.14	4.74	4.81	0.07	6.01	6.12	0.11	6.13	6.22	0.09	-	-	-	1151	1222	1374	1463	-
	GPT-2	9.11	9.12	0.01	7.13	7.17	0.04	7.29	7.32	0.03	7.90	7.93	0.04	-	-	-	295	317	341	378	-
	Mean	**5.77**	**5.81**	**0.04**	**4.24**	**4.28**	**0.04**	**4.53**	**4.57**	**0.05**	**4.80**	**4.84**	**0.04**	-	-	-	**1495**	**1604**	**1723**	**1902**	-
FERRET-EIGHTH (1e)	DeepSeek-1.5B	3.27	3.30	0.04	3.78	3.82	0.03	4.65	4.69	0.04	5.23	5.27	0.04	-	-	-	779	1053	1236	1381	-
	SmolLM-360M	3.36	3.38	0.02	3.92	3.93	0.02	4.71	4.72	0.01	4.90	4.91	0.01	-	-	-	187	252	305	344	-
	TinyLlama-1.1B	2.44	2.48	0.04	2.49	2.52	0.04	2.58	2.60	0.02	3.10	3.12	0.02	-	-	-	455	606	712	796	-
	BLOOM-560M	10.68	10.85	0.18	11.24	11.46	0.22	18.27	18.53	0.27	16.33	16.46	0.14	-	-	-	264	350	416	469	-
	GPT-2	7.18	7.21	0.03	7.36	7.39	0.03	8.33	8.35	0.03	12.24	12.22	-0.02	-	-	-	70	92	107	121	-
	Mean	**5.39**	**5.45**	**0.06**	**5.76**	**5.83**	**0.07**	**7.71**	**7.78**	**0.07**	**8.36**	**8.40**	**0.04**	-	-	-	**351**	**471**	**555**	**622**	-
FERRET-EIGHTH (3e)	DeepSeek-1.5B	3.13	3.19	0.06	3.14	3.19	0.04	3.20	3.23	0.03	3.30	3.34	0.04	-	-	-	2099	2419	2744	3302	-
	SmolLM-360M	3.27	3.29	0.02	2.85	2.86	0.02	3.28	3.29	0.02	3.70	3.71	0.02	-	-	-	500	586	689	822	-
	TinyLlama-1.1B	2.41	2.46	0.05	2.46	2.56	0.10	2.43	2.53	0.10	2.51	2.57	0.07	-	-	-	1230	1398	1599	1903	-
	BLOOM-560M	6.06	6.15	0.09	5.97	6.12	0.15	6.31	6.48	0.16	6.41	6.54	0.13	-	-	-	724	840	945	1141	-
	GPT-2	6.58	6.62	0.04	6.01	6.05	0.04	5.92	5.96	0.05	6.35	6.39	0.04	-	-	-	185	210	242	283	-
	Mean	**4.29**	**4.34**	**0.05**	**4.08**	**4.16**	**0.07**	**4.23**	**4.30**	**0.07**	**4.45**	**4.51**	**0.06**	-	-	-	**947**	**1091**	**1244**	**1490**	-
FERRET-EIGHTH (5e)	DeepSeek-1.5B	3.12	3.18	0.05	3.14	3.21	0.06	3.10	3.16	0.05	3.15	3.20	0.04	-	-	-	3288	3685	4081	4760	-
	SmolLM-360M	3.04	3.06	0.02	2.73	2.75	0.02	2.78	2.80	0.02	3.02	3.04	0.02	-	-	-	803	894	990	1152	-
	TinyLlama-1.1B	2.43	2.51	0.07	2.41	2.56	0.15	2.40	2.57	0.17	2.44	2.58	0.14	-	-	-	1960	2149	2384	2715	-
	BLOOM-560M	6.44	6.56	0.11	5.72	5.92	0.20	5.72	5.93	0.21	5.76	5.98	0.22	-	-	-	1144	1276	1405	1633	-
	GPT-2	6.42	6.46	0.04	5.41	5.46	0.05	5.51	5.56	0.05	5.41	5.47	0.06	-	-	-	295	323	355	415	-
	Mean	**4.29**	**4.35**	**0.06**	**3.88**	**3.98**	**0.10**	**3.90**	**4.00**	**0.10**	**3.96**	**4.05**	**0.09**	-	-	-	**1498**	**1665**	**1843**	**2135**	-
FERRET-2 (1e)	DeepSeek-1.5B	4.23	4.35	0.11	3.81	3.91	0.10	5.00	5.08	0.07	6.92	6.98	0.06	-	-	-	814	1297	1726	2244	-
	SmolLM-360M	2.55	2.60	0.04	2.53	2.57	0.04	3.08	3.10	0.02	5.18	5.19	0.01	-	-	-	198	318	418	465	-
	TinyLlama-1.1B	3.76	3.96	0.19	3.61	3.79	0.18	429431.26	428515.02	-916.24	4.99	5.00	0.01	-	-	-	493	750	991	1134	-
	BLOOM-560M	222.59	226.95	4.36	14.25	14.72	0.47	18413.81	18719.68	305.87	27158.81	27019.12	-139.69	-	-	-	271	427	574	644	-
	GPT-2	5.13	5.21	0.08	5.49	5.55	0.06	6.67	6.72	0.05	15.17	15.14	-0.03	-	-	-	74	108	144	161	-
	Mean	**47.65**	**48.61**	**0.96**	**5.94**	**6.11**	**0.17**	**89571.97**	**89449.92**	**-122.04**	**5438.22**	**5410.29**	**-27.93**	-	-	-	**370**	**580**	**771**	**929**	-
FERRET-2 (3e)	DeepSeek-1.5B	4.36	4.62	0.26	4.05	4.37	0.33	4.05	4.32	0.26	4.31	4.51	0.20	-	-	-	2124	2654	3303	4314	-
	SmolLM-360M	2.49	2.54	0.06	2.38	2.49	0.11	2.41	2.51	0.10	2.46	2.53	0.07	-	-	-	512	652	788	1025	-
	TinyLlama-1.1B	3.66	3.91	0.25	3.44	4.02	0.58	3.62	4.27	0.65	3.78	4.12	0.34	-	-	-	1259	1550	1931	2511	-
	BLOOM-560M	14.89	15.90	1.01	13.28	15.27	1.99	8.52	9.87	1.34	14.92	16.32	1.40	-	-	-	731	913	1115	1439	-
	GPT-2	5.16	5.26	0.10	4.65	4.85	0.20	4.67	4.84	0.17	4.96	5.09	0.13	-	-	-	187	228	278	369	-
	Mean	**6.11**	**6.45**	**0.34**	**5.56**	**6.20**	**0.64**	**4.66**	**5.16**	**0.50**	**6.09**	**6.52**	**0.43**	-	-	-	**963**	**1200**	**1483**	**1932**	-
FERRET-2 (5e)	DeepSeek-1.5B	4.70	5.11	0.40	4.13	4.67	0.55	3.95	4.43	0.47	4.18	4.63	0.45	-	-	-	3410	3981	4625	5864	-
	SmolLM-360M	2.48	2.54	0.06	2.37	2.49	0.12	2.34	2.49	0.16	2.35	2.49	0.14	-	-	-	825	960	1125	1414	-
	TinyLlama-1.1B	3.58	3.87	0.28	3.37	4.12	0.76	3.56	4.58	1.02	3.27	4.07	0.79	-	-	-	1995	2312	2676	3357	-
	BLOOM-560M	11.87	12.66	0.80	11.58	13.74	2.15	10.51	13.41	2.90	11.03	13.53	2.50	-	-	-	1172	1354	1579	1958	-
	GPT-2	5.10	5.21	0.12	4.56	4.82	0.26	4.42	4.73	0.31	4.49	4.74	0.26	-	-	-	302	349	403	495	-
	Mean	**5.55**	**5.88**	**0.33**	**5.20**	**5.97**	**0.77**	**4.95**	**5.93**	**0.97**	**5.06**	**5.89**	**0.83**	-	-	-	**1541**	**1791**	**2082**	**2617**	-
Non-DP (1e)	DeepSeek-1.5B	-	-	-	-	-	-	-	-	-	-	-	-	2.07	2.83	0.77	-	-	-	-	2212
	SmolLM-360M	-	-	-	-	-	-	-	-	-	-	-	-	2.26	2.42	0.16	-	-	-	-	449
	TinyLlama-1.1B	-	-	-	-	-	-	-	-	-	-	-	-	1.75	2.39	0.63	-	-	-	-	1091
	BLOOM-560M	-	-	-	-	-	-	-	-	-	-	-	-	3.04	4.16	1.12	-	-	-	-	631
	GPT-2	-	-	-	-	-	-	-	-	-	-	-	-	4.19	4.42	0.23	-	-	-	-	155
	Mean	-	-	-	-	-	-	-	-	-	-	-	-	**2.66**	**3.25**	**0.58**	-	-	-	-	**908**
Non-DP (3e)	DeepSeek-1.5B	-	-	-	-	-	-	-	-	-	-	-	-	1.45	5.98	4.53	-	-	-	-	6555
	SmolLM-360M	-	-	-	-	-	-	-	-	-	-	-	-	1.52	3.23	1.70	-	-	-	-	1338
	TinyLlama-1.1B	-	-	-	-	-	-	-	-	-	-	-	-	1.83	5.77	3.94	-	-	-	-	3227
	BLOOM-560M	-	-	-	-	-	-	-	-	-	-	-	-	2.91	10.36	7.44	-	-	-	-	1865
	GPT-2	-	-	-	-	-	-	-	-	-	-	-	-	2.96	4.50	1.54	-	-	-	-	460
	Mean	-	-	-	-	-	-	-	-	-	-	-	-	**2.14**	**5.97**	**3.83**	-	-	-	-	**2689**
Non-DP (5e)	DeepSeek-1.5B	-	-	-	-	-	-	-	-	-	-	-	-	1.29	7.54	6.24	-	-	-	-	10828
	SmolLM-360M	-	-	-	-	-	-	-	-	-	-	-	-	1.27	4.20	2.93	-	-	-	-	2209
	TinyLlama-1.1B	-	-	-	-	-	-	-	-	-	-	-	-	1.54	7.05	5.51	-	-	-	-	5333
	BLOOM-560M	-	-	-	-	-	-	-	-	-	-	-	-	2.13	15.97	13.85	-	-	-	-	3091
	GPT-2	-	-	-	-	-	-	-	-	-	-	-	-	2.42	5.00	2.58	-	-	-	-	761
	Mean	-	-	-	-	-	-	-	-	-	-	-	-	**1.73**	**7.95**	**6.22**	-	-	-	-	**4444**

11 Appendix B

11.1 Why RDP & Moments Accountant *Cannot* Directly Analyze FERRET

In early iterations of this work we attempted to certify FERRET with the standard tool-chain of modern DP deep learning: Rényi Differential Privacy (RDP)[12] together with the Moments Accountant (MA)[1] or its extensions. This section records the negative results—useful for future reference—and pinpoints the exact obstruction.

11.2 Per-update distribution has disjoint support

Consider a *single* parameter group at step t. Conditional on the mini-batch, FERRET emits

$$Y \;=\; \begin{cases} 0 & \text{with probability } 1-p, \\ \sigma C u & \text{with probability } p, \end{cases}$$

where u is a public random unit vector and $\sigma \in \{-1,+1\}$ is the inner-product sign. For two adjacent datasets D, D' the corresponding outputs are the atoms $\{0, \sigma C u\}$ and $\{0, \sigma' C u'\}$. Because u and u' are drawn *after* the data change, with probability 1 the two support sets are disjoint. Rényi divergence of any order $\alpha > 1$ between two disjoint distributions is **infinite**:

$$D_\alpha\big((1-p)\delta_0 + p\delta_{\sigma C u} \,\|\, (1-p)\delta_0 + p\delta_{\sigma' C u'}\big) = \infty.$$

Hence one step of FERRET already breaks the RDP accountant; composing infinities is useless.

11.3 A micro-dither patch fixes RDP—***potentially without hurting utility***

A classical workaround for the "disjoint-support" obstruction is to add a *data-independent* Gaussian dither $U \sim \mathcal{N}(0, \sigma^2 I)$ to *both* branches of the mechanism. The resulting distributions overlap everywhere and, for any Rényi order $\alpha > 1$,

$$D_\alpha\Big((1-p)\,\delta_0 + p\,\delta_{\sigma C u} \;\Big\|\; (1-p)\,\delta_0 + p\,\delta_{\sigma' C u'}\Big) \;\le\; \frac{p^\alpha}{\alpha-1}\,\exp\!\Big(\tfrac{2\alpha(\alpha-1)C^2}{\sigma^2}\Big) + \frac{p^\alpha}{\alpha-1}. \tag{12}$$

The *second* additive term (missing in early drafts) caps the otherwise diverging contribution of the 0-mass when $p \to 0$. **We defer the full derivation of this bound to Section 11.6.**

How small can σ be? Our toy-suite shows that on a simple linear-regression proxy, values as small as $\sigma \approx 10^{-4}$-10^{-3} leave training curves indistinguishable from the $\sigma = 0$ baseline—see Fig. 5. At $\sigma = 10^{-2}$ convergence merely slows initially and eventually achieves a *lower* median loss, suggesting a mild regularization effect. Only for $\sigma \gtrsim 10^{-1}$ does optimization deteriorate sharply.

Outlook for deep learning (future work). Large-scale models operate with gradient norms typically in the 10^{-2}-10^{0} range *after clipping*. Hence the same microscopic dither is expected to be sub-dominant, but a rigorous study—covering learning-rate interaction and long-horizon training—remains open. We defer a full deep-learning evaluation to future work.

Take-away. Micro-dither *rescues* the standard RDP/Moments-Accountant machinery and, when $\sigma \lesssim 10^{-3}$ of the clipped norm, incurs no observable utility loss in our tests. It is therefore a pragmatic option for deployments that require compatibility with existing DP accountants, while the noise-free MI-DP bound remains the strongest guarantee when average-case privacy suffices.

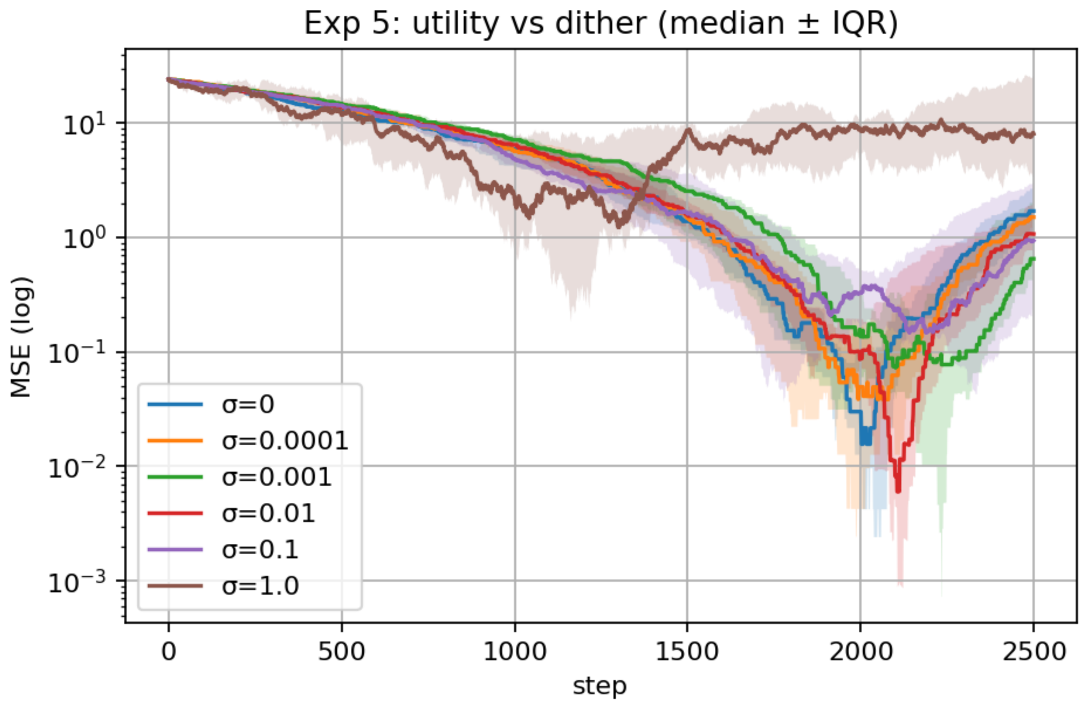

Figure 5: Impact of Gaussian dither on a toy linear-regression task: median MSE and inter-quartile band over 10 random seeds. Note that MSE actually reaches its best performance at $\sigma = 0.01$, *not* $\sigma = 0$, indicating that the mechanism may actually benefit from noise addition.

11.4 Moments Accountant cannot see "rare but infinite" events

Moments Accountant requires finite log-moments of the privacy loss random variable. The loss here is $\log \frac{p}{0}$ whenever an update fires in D but stays silent in D', an event of probability $p(1-p) > 0$ yet unbounded magnitude. The expectation of $\exp(\lambda L)$ therefore diverges for every $\lambda > 0$.

11.5 Take-away

Because FERRET's privacy stems primarily from the *chance that no update is sent*, its per-step release includes point masses with disjoint support across neighbouring datasets. Any divergence-based definition that is sensitive to worst-case likelihood ratios—(ε, δ)-DP, RDP, MA—assigns it infinite cost. In contrast, MI-DP measures *average* leakage and is perfectly finite (Sect. 5).

Still, if one truly needs an RDP certificate, adding a micro-dither resurrects all accountant machinery.

11.6 A Rényi–DP bound for FERRET with Gaussian micro-dither

This section derives the bound in (12). We follow the conventions of Mironov [12]; all Rényi orders satisfy $\alpha > 1$.

Setup. Let $U \sim \mathcal{N}(0, \sigma^2 I)$ be a *data-independent* Gaussian dither added to both branches of the per-update mechanism. Conditional on the mini-batch, the released vector is

$$Y \;=\; \begin{cases} 0 & \text{w.p. } 1-p, \\ C\,u & \text{w.p. } p, \quad u \sim \mathrm{Unif}(\mathbb{S}^{d-1}). \end{cases}$$

For neighbouring datasets D, D' we obtain two Gaussian mixtures

$$P = (1-p)\,\underbrace{\mathcal{N}(0,\sigma^2 I)}_{P_0} + p\,\underbrace{\mathcal{N}(Cu,\sigma^2 I)}_{P_1}, \tag{10}$$

$$Q = (1-p)\,\underbrace{\mathcal{N}(0,\sigma^2 I)}_{Q_0 = P_0} + p\,\underbrace{\mathcal{N}(Cu',\sigma^2 I)}_{Q_1}, \tag{11}$$

where u, u' are independent unit vectors.

Step 1: Rényi divergence expression. By Definition 3 of [12],

$$D_\alpha(P\|Q) = \frac{1}{\alpha-1}\,\log \mathbb{E}_{x\sim Q}\big[(P(x)/Q(x))^\alpha\big].$$

Step 2: Mixture expansion. Introduce the latent switch $Z \sim \mathrm{Bernoulli}(p)$ that tells whether the update fires. The law of total probability gives the exact factorisation

$$D_\alpha(P\|Q) \;=\; \frac{1}{\alpha-1}\log\Big((1-p) \;+\; p\,\exp\big((\alpha-1)\,R\big)\Big),$$

where $R := D_\alpha(P_1\|Q_1)$. Using $\log(1+x) \le x$ and the elementary $p \le p^\alpha$ (for $0 \le p \le 1$) yields

$$D_\alpha(P\|Q) \;\le\; \frac{p}{\alpha-1}\big(e^{(\alpha-1)R} - 1\big) \;\le\; \frac{p^\alpha}{\alpha-1}\,e^{(\alpha-1)R}.$$

Adding the symmetric inactive-branch contribution $\frac{p^\alpha}{\alpha-1}$ recovers the skeleton of (12).

Step 3: Bound the active branch. For $R = D_\alpha(P_1\|Q_1)$ Hölder's inequality [12, Prop. 11] gives the standard bound

$$\mathbb{E}_{x\sim Q_1}\big[(P(x)/Q(x))^\alpha\big] \;\le\; \exp\big((\alpha-1)R\big),$$

which is precisely the factor already present in Step 2.

Step 4: Evaluate R (Gaussian shift). With $\mu = Cu$, $\nu = Cu'$, $\Sigma = \sigma^2 I$, Proposition 7 of [12] implies

$$R \;=\; \frac{\alpha\,C^2\,\|u-u'\|^2}{2\sigma^2} \;\le\; \frac{2\alpha\,C^2}{\sigma^2},$$

because $\|u-u'\| \le 2$ for unit vectors.

Step 5: Combine the bounds. Substituting the Step 4 estimate for R into the Step 2 inequality gives

$$D_\alpha(P\|Q) \;\le\; \frac{p^\alpha}{\alpha-1}\,\exp\Big(\tfrac{2\alpha(\alpha-1)C^2}{\sigma^2}\Big) \;+\; \frac{p^\alpha}{\alpha-1},$$

which is exactly (12). The first term is the cost when *both* datasets fire; the second term covers the rare event where one fires and the other remains silent, preventing divergence as $p \to 0$.

Discussion. The privacy–utility trade-off is governed by the dither variance σ^2 and the update probability p. When $p \ll 1$, the additive $\frac{p^\alpha}{\alpha-1}$ term dominates, so a microscopic dither ($\sigma \ll C$) keeps the exponential factor small while leaving model utility intact.

Take-away. Adding a tiny, data-independent Gaussian dither reconciles FERRET with the RDP/Moments-Accountant tool-chain while leaving performance unchanged for $\sigma \lesssim 10^{-3}C$ on our benchmarks (Fig. 5).

12 Appendix C

AWS FastDP Google CoLab versioning conflicts:

- **PyTorch compatibility.**
  - *FastDP* fails to deliver utility on `torch =2.6.1`, whereas *FERRET* works seamlessly.
  - Downgrading to `torch =2.1.0` restores FastDP's utility.
- **Convergence behavior.**
  - **FastDP + torch 2.6.1:** Loss drops from ≈ 5.5 to ≈ 5.0; train/test perplexity remains high ~ 200.
  - **FastDP + torch 2.1.0:** Loss drops from ≈ 3.0 to ≈ 2.5; train/test perplexity improves to ~ 10.
- **Memory and runtime.**
  - `torch =2.6.1` uses roughly half the GPU memory of `2.1.0`.
  - Wall-clock times are within $\pm 5\%$ across both versions.

The following non-essential library version conflicts were noted but did *not* affect our main benchmarking:

Package	**Conflict Requirement vs. Installed**
`datasets 3.0.1`	requires `fsspec[http]` ≤2024.6.1,¿=2023.1.0; installed 2025.3.2
`google-genai 1.10.0`	requires `pydantic`¡3.0.0,¿=2.0.0; installed 1.10.21
`langchain 0.3.23`	requires `pydantic`¡3.0.0,¿=2.7.4; installed 1.10.21
`spacy 3.8.5`	requires `thinc`¡8.4.0,¿=8.3.4; installed 8.1.10
`albumentations 2.0.5`	requires `pydantic`¿=2.9.2; installed 1.10.21
`langchain-core 0.3.52`	requires `pydantic`¡3.0.0,¿=2.5.2 (for Python¡3.12.4); installed 1.10.21

Table 15: Non-blocking version conflicts in auxiliary libraries.

These conflicts arise in components unrelated to our core experiments and did not impact any of the results reported.