Unintended Memorization of Sensitive Information in Fine-Tuned Language Models

Marton Szep[1,2*†], Jorge Marin Ruiz[1*], Georgios Kaissis[2], Paulina Seidl[1], Rüdiger von Eisenhart-Rothe[1], Florian Hinterwimmer[1,2], Daniel Rueckert[2,3,4]

[1]Department of Orthopaedics and Sports Orthopaedics, TUM University Hospital
[2]Chair for AI in Healthcare and Medicine, Technical University of Munich (TUM) and TUM University Hospital
[3]Munich Center for Machine Learning (MCML)
[4]Department of Computing, Imperial College London

Abstract

Fine-tuning Large Language Models (LLMs) on sensitive datasets carries a substantial risk of unintended memorization and leakage of Personally Identifiable Information (PII), which can violate privacy regulations and compromise individual safety. In this work, we systematically investigate a critical and underexplored vulnerability: the exposure of PII that appears only in model inputs, not in training targets. Using both synthetic and real-world datasets, we design controlled extraction probes to quantify unintended PII memorization and study how factors such as language, PII frequency, task type, and model size influence memorization behavior. We further benchmark four privacy-preserving approaches including differential privacy, machine unlearning, regularization, and preference alignment, evaluating their trade-offs between privacy and task performance. Our results show that post-training methods generally provide more consistent privacy–utility trade-offs, while differential privacy achieves strong reduction in leakage in specific settings, although it can introduce training instability. These findings highlight the persistent challenge of memorization in fine-tuned LLMs and emphasize the need for robust, scalable privacy-preserving techniques.

1 Introduction and Related Work

Large Language Models (LLMs) achieve state-of-the-art performance across numerous natural language processing tasks, but their vast capacity and data-hungry training regimes raise serious privacy concerns. Most notably, LLMs can memorize training samples even if seen only once (Carlini et al., 2021). While some memorization supports generalization in long-tailed data distributions (Feldman and Zhang, 2020), verbatim token-level memorization of Personally Identifiable Information (PII) poses significant privacy risks.

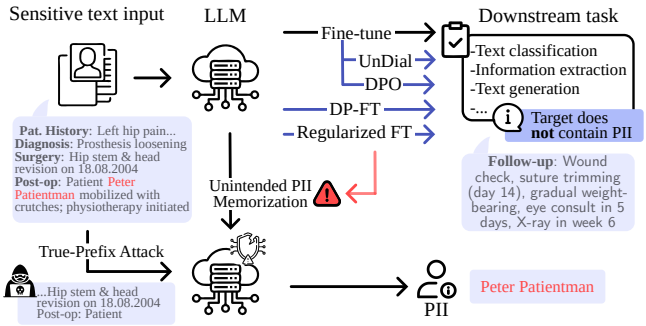


Figure 1: Overview of our experiment setup depicting the unintended PII memorization scenario, our attack, and fine-tuning approaches.

Prior studies have analyzed memorization dynamics in LLMs during both pre-training (PT) and fine-tuning (FT) (Morris et al., 2025; Carlini et al., 2021; Hu et al., 2022). For example, Carlini et al. (2022) examined how model size, data duplication, and prompt length influence memorization during PT, and Zeng et al. (2024) studied task-specific memorization during FT. These works primarily focus on task-relevant data. In practice, however, FT often (inadvertently) involves inputs containing sensitive PII that are *unrelated to the task output*, such as names or medical records. Despite growing attention to LLM privacy, the specific risk of *unintended PII memorization*, where PII appears only in inputs and is irrelevant to the downstream task, remains largely unexplored.

Recent studies have highlighted PII leakage in realistic deployment scenarios, including black-box probing and adversarial API querying (Nakka et al.,

*These authors contributed equally.
†Corresponding author: marton.szep@tum.de. Code is available at https://github.com/martonszep/llm-pii-leak.

Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics
Volume 1: Long Papers, pages 6461–6480

2024b,a; Lukas et al., 2023). Yet, no prior work systematically isolates input-only PII memorization or compares mitigation strategies across methods. In this work, we present a comprehensive study addressing this gap. Specifically, we:

- Define and formalize the problem of input-only PII memorization, distinguishing it from general memorization or task-relevant PII usage;
- Quantify memorization using synthetic and real-world datasets in realistic deployment scenarios;
- Analyze key factors influencing memorization severity, including language, PII repetition, model capacity, prefix length, and downstream task type;
- Benchmark four common mitigation strategies, assessing their privacy-utility tradeoffs: differential privacy, regularization, machine unlearning, and preference alignment.

To the best of our knowledge, this is the first comprehensive study focused on unintended PII memorization in LLM fine-tuning, highlighting challenges and considerations for privacy-preserving model deployment.

2 Methodology

2.1 Unintended, Input-only PII Memorization

We define unintended PII memorization as the phenomenon in which a language model fine-tuned on sensitive text data, such as electronic health records (EHRs), internalizes PII that is not part of its intended output (i.e., unrelated to the downstream task). This is distinct from memorization during pre-training, where large corpora might contain public or semi-public PII, and from targeted FT tasks, where PII is intentionally part of the model's output space.

Our work focuses on downstream tasks (classification, information extraction, medical follow-up planning) in which PII appears only in the inputs, not in the training targets. We adopt a realistic black-box threat model where adversaries access the model only via input-output queries (e.g., API calls). We assume a worst-case scenario where attackers have partial access to the FT dataset (e.g. anonymized EHRs) and can craft adversarial prompts accordingly (Carlini et al., 2022; Nakka et al., 2024b).

True-Prefix Attack (TPA) is a method to probe memorization in autoregressive LLMs (Carlini et al., 2021). Given a true prefix c from the FT data immediately preceding a PII span s of N tokens, we say s is extractable if

$$s \leftarrow \underset{s' : |s'|=N}{\arg\max} f_\theta(s' \mid c). \tag{1}$$

where $f_\theta(s' \mid c)$ is the model's conditional probability distribution. With labeled PII spans, this attack is straightforward to construct and evaluate, providing an effective measure of model memorization. Throughout the paper, we use TPA as the main PII extraction method and also define an *enhanced* TPA variant, which adds the first character of the PII to the prefix. We also compare the true-prefix attack to other adversarial attacks employing more sophisticated prompting techniques in § A.3, and find that the extraction rate achieved by TPA serves as an approximate lower bound.

2.2 Mitigating memorization

We evaluate four prevalent training strategies aimed at reducing PII memorization during or after FT.

Differential Privacy (DP) is a widely used technique for protecting individual data privacy with mathematical guarantees (Kulynych et al., 2025; Dwork, 2006). It introduces noise into the gradient updates and limits individual sample influence, thus bounding sample-level memorization risk. DP has been extensively applied to both LLM pre-training and fine-tuning, providing verifiable guarantees, but at the cost of utility degradation and increased training complexity (Hoory et al., 2021; Li et al., 2021; Yu et al., 2021).

Machine Unlearning: UnDial (Dong et al., 2024) is a targeted unlearning method based on self-distillation. It constructs an adjusted target distribution by suppressing the logits of tokens associated with the information to be forgotten. By relying on self-distillation rather than directly maximizing loss on the forget set, UnDial mitigates training instability and reduces unintended degradation of general model performance, issues observed in earlier unlearning methods such as Gradient Ascent and Negative Preference Optimization (Fan et al., 2025; Shi et al., 2024).

Regularization Inspired by UnDial (Dong et al., 2024), we propose a regularization-based variant that integrates self-distillation into the FT loop. Specifically, we alternate between cross-entropy loss and a regularization loss focused on PII tokens. This focused UnDial loss is applied only on selected sensitive spans to discourage memorization.

Direct Preference Optimization (DPO) emerges as a computationally and data-efficient alternative to RLHF for aligning models' outputs with human preferences, such as privacy or helpfulness (Rafailov et al., 2023; Szep et al., 2024). We adapt DPO to discourage PII leakage by treating training examples that contain PII as *rejected* and their corresponding masked versions as *preferred*.

3 Experiments

In our experiments, we primarily focus on extracting memorized personal names, as they constitute the most heterogeneous and unstructured category of PII and defy rule-based detection using regular expressions. Our extended analysis across PII types in § A.4 reveals that while SFT universally exacerbates leakage across all PII types in synthetic data, it yields mixed results in real-world domains. Specifically, we find that fine-tuning can reduce the extraction of generic structured priors (e.g., dates, postal codes) compared to the base model, in contrast to the persistent memorization risks observed for unstructured identifiers.

3.1 Datasets

We use three datasets varying significantly in nature, task complexity, and objectives. The latter two are private medical datasets of German EHRs provided by the Orthopaedics Department of the University Hospital of the Technical University of Munich. For additional details on data and preprocessing, see Appendix B.

GretelAI-Financial (Watson et al., 2024) is a synthetic, multilingual NER dataset focused on PII. After preprocessing, it contains ~30k samples in 7 languages with 52 financial text classification labels, which we use for the downstream task.

Pathology reports consist of 2,553 German documents summarizing bone and soft-tissue tumor analyses collected between 2020 and 2023. Reports include rich domain-specific terminology (e.g., tumor dignity, entity, and intervention type) and manually annotated PII verified by medical professionals. The data was cleaned, normalized, and split into structured 5-field JSON records for information extraction.

Discharge Summary (DS) comprise 26,306 German clinical reports from 1996–2014 covering patient history, treatment, and follow-up plans. We extract the *Procedere* section to form a follow-up plan generation task and employ a multi-stage PII annotation pipeline combining regex, LLM-based detection, and manual verification to ensure comprehensive coverage of personal identifiers.

3.2 Privacy-preserving training

We quantify the PII memorization during vanilla fine-tuning and benchmark different privacy-preserving training methods. Further training details can be found in Appendix C.

Supervised Fine-Tuning We establish memorization baselines by primarily fine-tuning Llama 3.2 1B models (Grattafiori et al., 2024) using QLoRA ($r = 8$) in all linear layers over $10 - 25$ epochs (varying per dataset), until triggering early stopping. A cosine learning rate scheduler with linear warmup of 3% of steps is used. Hyperparameters are optimized only for downstream performance, without privacy considerations.

Differential Privacy Fine-Tuning We integrate (ε, δ)-DP into the QLoRA setup via Opacus' Privacy Engine (Yousefpour et al., 2022), using privacy budgets $\varepsilon \in [2, 8]$ and $\delta = 10^{-5}$. Hyperparameter choices follow Li et al. (2021) to maximize utility under DP constraints.

UnDial We apply UnDial to a disjoint subset of $17692\,(40\%)$, $1500\,(40\%)$, and $6000\,(20\%)$ person names in the GretelAI, Pathology, and DS datasets respectively; none of which overlap with the names extracted during our memorization assessment (see § 3.3). We use the same input-output structure for unlearning as for the TPA, with the PII being the unlearning target.

Regularization We apply regularization using focused UnDial (Dong et al., 2024) to compute the regularization loss only over the PII tokens, using the same PII subset as for unlearning.

DPO Following FT, we run DPO with a uniform system prompt instructing the model to withhold all PII. For each corpus, we slide a 150-token context window over sequences containing at least two PII within the following 20 tokens. We mask all PII in these 20-token spans to form the preferred response and use the original, unmasked text as the rejected response. The resulting datasets contain 1489 (GretelAI) and 5636 (DS) training samples.

3.3 Evaluation

We use task-specific evaluation metrics: accuracy for GretelAI, F1-score for Pathology, and

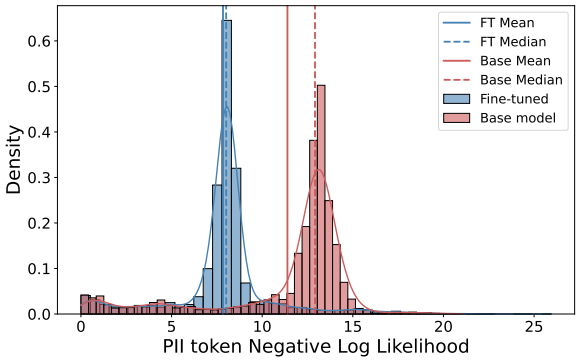


Figure 2: Distribution of per-token log-likelihoods for ground-truth PII completions.

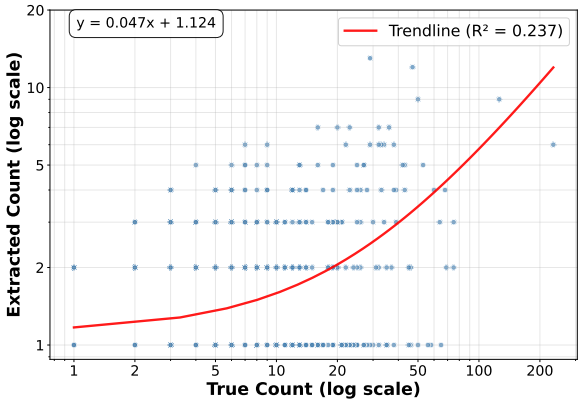


Figure 3: Relation of extracted PII counts (enhanced TPA, Cross-memorization) and true counts in the DS dataset for Llama 3.2 1B (seed 2431).

BERTScore-F1 for DS. In our PII extraction experiments, we adopt three evaluation settings:

1. *Greedy*: Greedy decoding ensures reproducibility and comparability.
2. *Sampling*: We perform TPA evaluation by sampling the models 32 times per prefix, setting the model temperature to 1.
3. *Cross-memorization*: we also evaluate by comparing generations to all PII of the same kind in the dataset (in addition to the ground-truth).

For TPA, generation is capped at 25 tokens following the prefix (50 tokens). For additional details about memorization and downstream task evaluation, we refer the reader to Appendix D.

4 Results

4.1 Understanding input-only PII memorization

Fine-tuned models are more confident in predicting PII tokens. Figure 2 shows the density of per-token negative log-likelihoods for the FT and base models over the same PII (names) in the DS dataset. The fine-tuned model's distribution mode is shifted substantially closer to zero and has significantly smaller variance compared to the base model. This indicates that FT has increased the model's confidence across PII tokens.

PII frequency is a poor predictor of unintended memorization. Our findings challenge the assumption that high frequency in the training data directly correlates with higher rates of unintended memorization (Carlini et al., 2022). To investigate this relationship, we plotted the true count of each PII instance in our DS training set against the count of its successful extraction from the Llama 3.2 1B model (Figure 3). The analysis shows a very weak positive linear relationship, with a trendline of $y = 0.047x + 1.124$ and a low coefficient of determination ($R^2 = 0.237$). This demonstrates that the repetition of a PII token is a poor predictor, explaining less than 24% of the variance in whether it is memorized and extracted. We hypothesize that memorization is more heavily influenced by the PII's textual context and its utility to the downstream task. For example, PII located in document headers (some of the most frequent yet task-irrelevant tokens in the DS corpus) were only memorized with an order of magnitude larger learning rate, further decoupling raw frequency from memorization risk. This hypothesis is also reflected by our results on the Pathology dataset (Table 2), where PII leakage was significantly lower. We attribute this to its smaller size, sparser PII distribution. We investigate the influence of the downstream task on memorization in greater detail in Appendix A.1.

Language affects memorization. To examine whether memorization behavior depends on language, we evaluated PII extraction rates across seven languages. The models were fine-tuned jointly on the multilingual GretelAI-Financial dataset, and this process was repeated three times with different random seeds. A repeated-measures ANOVA (Girden, 1992), treating model seed as a random factor and language as a within-subject variable, revealed a highly significant main effect of language on extraction rate ($F(6, 12) = 61.13$, $p = 2.68 \times 10^{-8}$). This confirms that, even when trained on the same data, the models exhibit systematic cross-lingual variation in memorization.

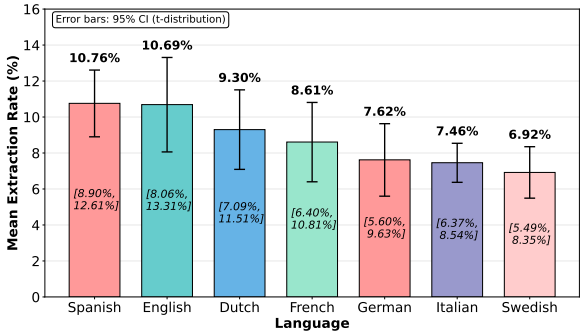


Figure 4: PII extraction success ratio across languages for GretelAI-Financial with Llama 3.2 1B. Error bars represent 95% confidence intervals over 3 random seeds.

Figure 4 shows the mean extraction rate with 95% confidence intervals per language.

Post-hoc Tukey HSD comparisons (Tukey, 1949) indicated that English and Spanish have significantly higher extraction rates than German, Italian, and Swedish ($p < 0.01$), while Dutch and French occupy an intermediate range without significant differences from either group (Table 7). These differences, although statistically robust, are moderate in magnitude (typically 2–4 percentage points). Finally, we emphasize that despite extensive filtering and curation, these findings are derived from synthetic data, which may limit their generalizability.

Effect of prefix length in TPA. Following Carlini et al. (2022), we investigated the effect of prefix token length on attack success and found that, consistent with their work, effectiveness increases sharply with longer prefixes, particularly between 25 and 50 tokens, with only marginal gains beyond this point. Based on this observation, we standardized a 50-token prefix for most evaluations. However, a more granular analysis (Figure 8) reveals complex, dataset-specific trends. While fine-tuned models generally show a logarithmic increase in success with prefix length, the pretrained model's performance varies: on the GretelAI dataset, success improves with prefixes up to 100 tokens, whereas on the DS dataset, it plateaus between 50–100 tokens and can even decrease with longer contexts. These patterns align partially with Carlini et al. (2022) but suggest possible dataset-specific trends in unintended PII memorization.

Scaling effects on task performance and PII leakage. Figure 5 and Table 1 compare task performance and PII extraction behavior across multiple architectures and parameter scales. Larger models demonstrate higher baseline capacity to reveal PII, even without fine-tuning, suggesting that scaling amplifies inherent memorization capabilities. However, unlike smaller models (<4B), fine-tuning does not consistently increase the raw count of extractable PII for 8B–12B models. Still, the absolute amount of reproduced PII rises with model size, indicating that larger models may replicate sensitive content more reliably once exposed. Across architectures, input-only memorization remains persistent but varies by model size and evaluation mode: Llama 3.2 3B shows the strongest increase after fine-tuning, followed by Llama 3.2 1B and Llama 3.1 8B; similarly, Gemma-3 1B exhibits a greater rise than both Gemma-3 4B and Gemma-3 12B. Qwen-3 1.7B, in turn, appears particularly vulnerable to regular true-prefix attacks after fine-tuning. These results collectively highlight that memorization risk is influenced not only by model size but also by architecture, with fine-tuning effects varying substantially across models.

Further, we investigated model scaling at the level of the trainable parameters. As shown in Table 5, an eight-fold increase in LoRA rank does **not** raise the total number of PII extractions, which remains constant under both query-per-prefix settings. However, the number of unique PII instances increases across both, indicating that while additional parameters do not elevate overall memorization volume, they expand its diversity by exposing a broader range of unique identifiers.

4.2 Mitigating unintended PII memorization

Post-training methods offer robustness, though DP can be competitive in specific settings. Across datasets (Table 2), post-training mitigation methods such as DPO and UnDial generally yield more consistent privacy–utility trade-offs and are more robust to hyperparameter variation. They are also less resource-intensive than preventive techniques like DP and regularization. However, DP shows strong privacy potential in specific scenarios. In the DS task, it reduces cross-memorization by over 60%, the highest among all methods, even without using seed PII data. Yet, DP remains unstable to train, often requiring larger batch sizes, higher learning rates, and longer training, with results varying substantially across runs. We also observe that DP models occasionally produce repetitive outputs under TPA, indicating possible degradation in generation quality. Regularization suffers from conflicting training objectives, preserv-

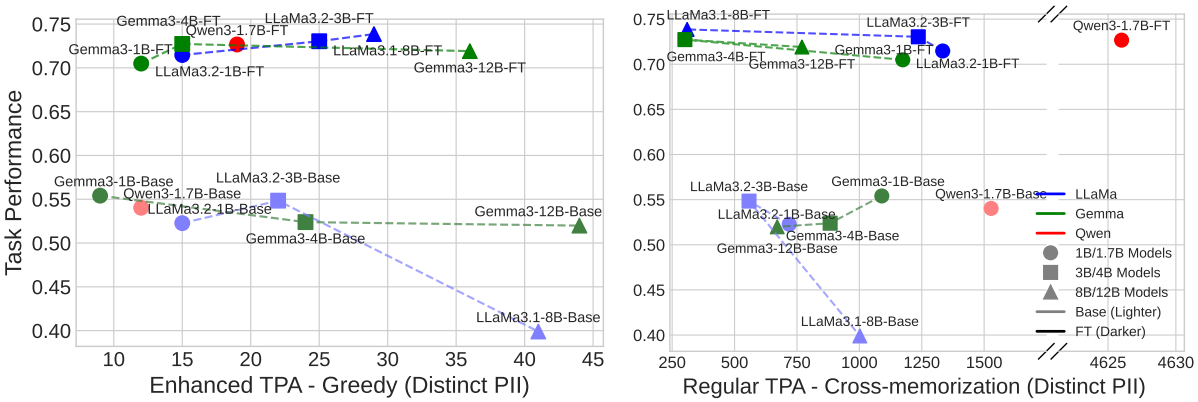


Figure 5: Effect of different model architectures and sizes on task performance and PII memorization on the Discharge Summary dataset. See Table 1 for the complete breakdown of PII extraction results.

Model	Task Performance↑	Regular TPA Greedy↓	Regular TPA Cross↓	Enhanced TPA Greedy↓	Enhanced TPA Cross↓
LLaMa3.2-1B-Base	0.5227	1 (1)	1940 (719)	25 (15)	9638 (3974)
LLaMa3.2-1B-FT	0.7147	**1 (1)**	1604 (1334)	91 (15)	11754 (4453)
LLaMa3.2-3B-Base	0.5484	6 (6)	945 (558)	40 (22)	9112 (3221)
LLaMa3.2-3B-FT	0.7304	6 (5)	4803 (1236)	57 (25)	13920 (3560)
LLaMa3.1-8B-Base	0.3990	13 (13)	1781 (1002)	83 (41)	8379 (4083)
LLaMa3.1-8B-FT	**0.7386**	8 (7)	5469 (310)	48 (29)	7564 (1281)
Gemma3-1B-Base	0.5541	4 (3)	1406 (1090)	16 (9)	4885 (3349)
Gemma3-1B-FT	0.7049	3 (3)	1539 (1174)	**21 (12)**	7143 (3105)
Gemma3-4B-Base	0.5239	12 (7)	1234 (883)	38 (24)	2338 (1352)
Gemma3-4B-FT	0.7273	19 (13)	**499 (301)**	24 (15)	**5925 (1080)**
Gemma3-12B-Base	0.5199	51 (39)	1183 (671)	90 (44)	8934 (2774)
Gemma3-12B-FT	0.7191	40 (32)	1437 (770)	70 (36)	6292 (3098)
Qwen3-1.7B-Base	0.5403	3 (3)	3272 (1528)	15 (12)	10633 (6466)
Qwen3-1.7B-FT	0.7267	7 (3)	9185 (4626)	47 (19)	11650 (5505)

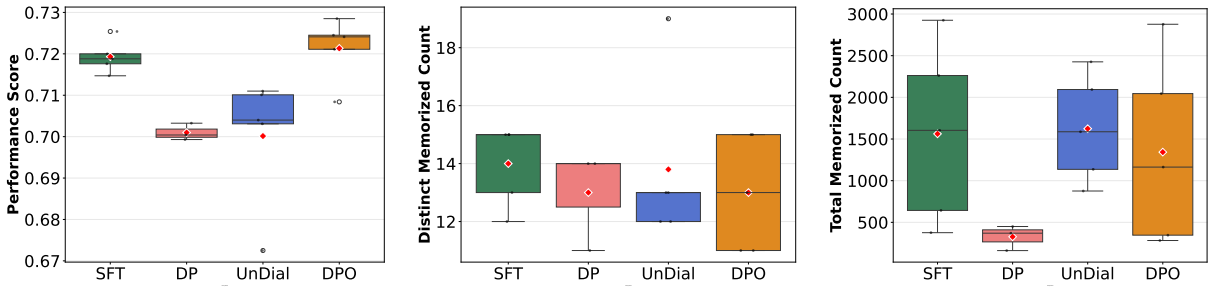
Table 1: Comparison of task performance and PII memorization across different models. Each cell reports *Total PII Instances (Unique PII Entities)* for greedy decoding and cross-memorization conditions. Best scores are highlighted in bold for each FT model.

ing task performance but retaining more PII. Unlearning and alignment methods are sensitive to the quality and size of the seed set, requiring careful tuning to balance effectiveness and utility. However, they consistently and considerably outperform other methods in our sampling-based benchmark. Overall, while DP can outperform in isolated cases for greedy decoding and cross-memorization, post-training methods excel in robustness against sampling-based TPA and offer quicker and more stable training.

Combining training approaches does not bring out the best of both worlds. We evaluated post-training DPO on a DP baseline (DP-ϵ6 + DPO in Table 2). The hybrid largely preserves DP's strengths under greedy decoding (both achieve zero distinct PII) and slightly improves cross-memorization relative to DP alone. However, it does not inherit DPO's advantage under sampling-based attacks: sampling counts are higher than DPO and are comparable to or worse than DP in the enhanced TPA, while task performance is slightly reduced compared to standalone DPO. Thus, the hybrid offers DP-like robustness in greedy and cross-memorization scenarios but fails to fully realize DPO's sampling-based protections, suggesting that more careful integration or tuning is required to achieve the complementary benefits. This is further reflected in the training dynamics (Figure 9): while DP+DPO exhibits stable convergence with loss curves similar in shape to DPO, it converges to slightly higher training and validation losses, and shows less pronounced improvements despite reduced noise, suggesting that DP pre-training constrains the optimization landscape in a way that limits the downstream gains from DPO.

	Method	Task Performance↑	Regular True-Prefix Attack			Enhanced True-Prefix Attack		
			Greedy↓	Sampling↓	Cross↓	Greedy↓	Sampling↓	Cross↓
GretelAI - Financial	Base	0.1208	3402 (1758)	-	-	-	-	-
	SFT	0.8717	3601 (1720)	-	-	-	-	-
	DP-ϵ2	0.6616	3304 (1654)	-	-	-	-	-
	DP-ϵ6	0.7484	3563 (1767)	-	-	-	-	-
	UnDial-40%	0.7621	2717 (1323)	-	-	-	-	-
	Reg-40%	**0.8112**	3297 (1534)	-	-	-	-	-
	DPO-β0.01	0.7924	**2616 (1167)**	-	-	-	-	-
Pathology	Base	0.2889	0 (0)	72 (56)	6 (4)	0 (0)	68 (61)	7 (6)
	SFT	0.8621	0 (0)	63 (51)	11 (7)	0 (0)	69 (57)	10 (8)
	DP-ϵ6	0.5513	0 (0)	60 (50)	9 (6)	0 (0)	63 **(52)**	7 (7)
	UnDial-40%	**0.7189**	**0 (0)**	**56 (50)**	**6 (5)**	**0 (0)**	**60** (53)	**7 (5)**
Discharge Summary	Base	0.5227	1 (1)	82 (37)	1940 (719)	25 (15)	815 (205)	9638 (3974)
	SFT	0.7147	1 (1)	71 (46)	1604 (1334)	91 (15)	849 (165)	11754 (4453)
	DP-ϵ2	0.6906	0 (0)	-	1143 (733)	43 (16)	-	17405 (5994)
	DP-ϵ6	0.6993	**0 (0)**	91 (44)	161 (154)	**30 (11)**	1109 (201)	**5624 (1589)**
	UnDial-20%	0.6725	1 (1)	43 **(24)**	1587 (1103)	31 (13)	752 (139)	9456 (3593)
	Reg-20%	0.6770	2006 (17)	-	5388 (2227)	6841 (142)	-	17102 (6601)
	DPO-β0.01	**0.7084**	1 (1)	**42** (33)	1163 (1009)	31 (13)	**733 (115)**	6298 (2860)
	DP-ϵ6 + DPO	0.6820	**0 (0)**	107 (53)	**155 (148)**	34 (13)	1098 (191)	6108 (1798)

Table 2: Comparison of PII (names) memorization and task performance across methods and datasets for the Llama 3.2 1B model. Each cell represents *Total PII Instances (Unique PII Entities)*. Columns correspond to greedy decoding, sampling with temperature 1.0 over 32 runs, and cross-memorization. Best scores are highlighted in bold.



(a) Task Performance (b) Enhanced TPA - Greedy (c) Regular TPA - Cross-memorization

Figure 6: Variance across multiple training runs in terms of (a) task performance and (b, c) distinct PII memorization on the Discharge Summary dataset. See Table 3 for mean and standard deviation values.

Stability and effectiveness across random seeds. Figure 6 and Table 3 summarize the multi-seed evaluation of task performance and PII extraction for the Llama 3.2 1B model. Differential privacy (DP) consistently yields the strongest reduction in PII leakage across both greedy and cross-memorization settings, with minimal performance degradation and low variance across seeds. DPO achieves the highest task accuracy but provides weaker leakage reduction in this setup. Nonetheless, single-seed evaluations (cf. Table 2) demonstrated that DPO and UnDial can effectively mitigate sampling-based leakage. Both methods exhibit moderate privacy improvements but higher variability, likely reflecting sensitivity to initialization and the limited amount of post-training data—a factor we leave for future investigation. Overall, DP remains the most stable approach, yet measurable leakage persists under enhanced attacks. Crucially, even the most effective methods achieve only around a 40% reduction in direct PII memorization, indicating substantial room for improvement.

Scaling effects on privacy-preserving fine-tuning. Table 4 compares privacy-preserving fine-tuning methods across 1B–8B Llama 3.2 models. Consistent with earlier findings, post-training approaches (DPO, UnDial) achieve the most favorable privacy–utility trade-off overall, with DP providing the strongest protection for the 1B. However, scaling up parameters alters this balance: while task performance improves, DP no longer

	Task Performance↑	Regular TPA		Enhanced TPA	
		Greedy↓	Cross↓	Greedy↓	Cross↓
SFT	0.7193 ± 0.0039	0.8 ± 0.8 (0.8 ± 0.8)	1562.0 ± 1072.5 (1081.2 ± 668.6)	45.6 ± 25.6 (14.0 ± 1.4)	9835.4 ± 2813.2 (3640.2 ± 1099.8)
DP	0.7010 ± 0.0021	$\mathbf{0.0 \pm 0.0}$ $\mathbf{(0.0 \pm 0.0)}$	$\mathbf{327.0 \pm 149.2}$ $\mathbf{(273.0 \pm 111.5)}$	$\mathbf{27.3 \pm 5.5}$ $\mathbf{(13.0 \pm 1.7)}$	7013.3 ± 2669.5 $\mathbf{(2318.0 \pm 1280.0)}$
UnDial	0.7001 ± 0.0158	1.4 ± 1.5 (1.4 ± 1.5)	1631.0 ± 656.1 (1161.0 ± 322.5)	29.0 ± 4.4 (13.8 ± 2.9)	10065.8 ± 2247.8 (3871.8 ± 1200.4)
DPO	$\mathbf{0.7213 \pm 0.0077}$	1.6 ± 0.9 (1.6 ± 0.9)	1342.6 ± 1117.8 (1007.4 ± 831.8)	31.0 ± 5.9 $\mathbf{(13.0 \pm 2.0)}$	$\mathbf{6930.2 \pm 2672.8}$ (2512.0 ± 1054.7)

Table 3: Mean and standard deviation of PII (names) memorization and task performance across different training methods for Llama 3.2 1B on the DS dataset. Each cell reports *Total PII Instances (Unique PII Entities)* for greedy decoding and cross-memorization settings. Best scores are highlighted in bold. See Table 6 for the scores of the individual runs.

	Model	Task Performance↑	Regular True-Prefix Attack			Enhanced True-Prefix Attack	
			Greedy↓	Sampling↓	Cross↓	Greedy↓	Cross↓
Llama 3.2 1B	Base	0.5227	1 (1)	82 (37)	1940 (719)	25 (15)	9638 (3974)
	SFT	0.7147	1 (1)	71 (46)	1604 (1334)	91 (15)	11754 (4453)
	DP-ϵ6	0.6993	**0 (0)**	91 (44)	**161 (154)**	**30 (11)**	**5624 (1589)**
	UnDial-20%	0.6725	1 (1)	43 **(24)**	1587 (1103)	31 (13)	9456 (3593)
	DPO	**0.7084**	1 (1)	**42** (33)	1163 (1009)	31 (13)	6298 (2860)
Llama 3.2 3B	Base	0.5484	6 (6)	164 (77)	945 (558)	40 (22)	9112 (3221)
	SFT	0.7304	6 (5)	133 (50)	4803 (1236)	57 (25)	13920 (3560)
	DP-ϵ6	0.7003	**1 (1)**	85 (61)	1043 (511)	**23 (13)**	8735 (1134)
	UnDial-20%	0.7148	2 (2)	138 (74)	1025 (867)	37 (20)	13464 (3824)
	DPO	**0.7299**	4 (3)	**25 (8)**	**532 (457)**	26 **(13)**	**4632 (954)**
Llama 3.1 8B	Base	0.3990	13 (13)	478 (101)	1781 (1002)	83 (41)	8379 (4083)
	SFT	0.7386	8 (7)	226 (58)	5469 (310)	48 (29)	7564 (1281)
	DP-ϵ6	**0.7151**	13 (12)	264 (79)	1177 (558)	50 (29)	6594 (2135)
	UnDial-20%	0.6353	11 (11)	202 (82)	1195 (861)	51 (22)	11215 (2295)
	DPO	0.7011	**0 (0)**	**14 (7)**	**18 (18)**	**3 (3)**	**43 (42)**

Table 4: Scaling behavior of training methods over model sizes for the Discharge Summary dataset. Each cell reports *Total PII Instances (Unique PII Entities)*. Columns correspond to greedy decoding, sampling with temperature 1.0 over 32 runs, and cross-memorization. Best scores are highlighted in bold for each model.

prevents memorization to the same extent, especially under cross-memorization settings. In contrast, DPO proves markedly more effective for the larger model, substantially reducing both direct and cross-memorization leakage while maintaining near-optimal task performance. This suggests that as model capacity increases, preference-based post-training can better leverage alignment signals without amplifying memorized PII. Still, no method eliminates leakage entirely, highlighting persistent trade-offs between scalability, utility, and privacy.

5 Discussion

This work provides a systematic analysis of unintended PII memorization in fine-tuned language models. We identify key influencing factors and evaluate four mitigation strategies with varying trade-offs in privacy, utility, and stability. Fine-tuning on small, domain-specific datasets may lessen memorization but does not eliminate the risk. Post-training methods such as DPO and UnDial generally offer more consistent privacy–utility trade-offs. Meanwhile, differential privacy can provide stronger leakage reduction in some cases but remains unstable, highly sensitive to hyperparameters, and prone to utility degradation.

Unintended, input-only memorization is not only a technical challenge but also one with direct societal implications. If not mitigated, memorization risks could enable misuse, such as attempts to recover sensitive records from deployed models or the targeting of vulnerable populations, for exam-

ple patients in clinical contexts or speakers of minority languages whose data may be more uniquely identifiable within niche training sets. Our findings underscore the importance of carefully evaluating privacy safeguards before deploying fine-tuned models in sensitive domains.

A key open question is why current approaches, even when effective, still leave ample room for improvement. A deeper investigation into these mechanisms, along with the design of more robust and scalable defenses, remains an important direction for future work.

Limitations

Our study primarily examines parameter-efficient fine-tuning (QLoRA) on LLMs up to 12B parameters. While this extends beyond smaller-scale experiments, larger foundation models and reasoning-oriented architectures may exhibit different memorization behaviors and mitigation responses. Future work should investigate whether our findings generalize to these settings and to alternative parameter-efficient training methods, non-quantized models, or full-model fine-tuning.

Dataset availability and label quality remain another limitation. High-quality, large-scale corpora with explicit PII annotations are effectively unavailable due to privacy and ethical constraints. Consequently, we rely on (1) synthetic multilingual financial data, (2) a small, manually annotated private dataset, and (3) a larger private corpus with PII spans identified through a semi-automated pipeline (see Appendix B). While these sources provide complementary perspectives, reliance on synthetic data and automated annotation may introduce bias, and the absence of individual-level structuring (e.g., per-patient granularity) limits exploration of privacy-preserving approaches such as federated learning or user-level differential privacy. We view this gap as an opportunity for future dataset development, which would significantly strengthen the field.

Finally, while we provide a preliminary comparison of several adversarial extraction strategies in § A.3 (including both instruction-based and jailbreak attacks), we do not claim an exhaustive assessment of the adversarial landscape. Systematic evaluation of threats like white-box gradient leaks or highly optimized manual jailbreaks remains methodologically challenging and often requires extensive model-specific tuning. Our work primarily focuses on quantifying token-level memorization under standard inference conditions as a foundation, thereby motivating future, more comprehensive investigations into the adversarial robustness of these defense mechanisms.

Ethics Statement & Data Privacy

This study examines memorization risks in fine-tuned language models using datasets that include annotated personally identifiable information (PII) spans. Two private datasets with medical data were obtained from the University Hospital of the Technical University of Munich and used under approval of the local institutional ethics review board (2025-574_1-S-NP). The board determined that informed consent could be waived for this study.

The use of identifiable data was essential for this study's scientific objectives, as assessing the unintended memorization of PII in LLMs inherently requires access to real, identifiable references. Without such data, it would be impossible to evaluate whether models reproduce or leak sensitive information, which is a core research question of significant public and scientific relevance. Therefore, this work constitutes a justified exception in which the research interest demonstrably outweighs potential risks to individuals. No feasible alternative using synthetic or pre-anonymized data could address the same question with sufficient validity.

All processing took place under strict technical and organizational safeguards within a secure in-house research environment. No data were transferred outside institutional systems, and no model weights fine-tuned on sensitive material were released. A multi-stage detection process combining automated tools with manual verification was used to ensure accurate identification and handling of PII. Data preprocessing, anonymization procedures, and annotation workflows are detailed in Appendix B.

References

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. 2022. Quantifying Memorization Across Neural Language Models. In *The Eleventh International Conference on Learning Representations*.

Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. 2021.

Extracting Training Data from Large Language Models. In *30th USENIX Security Symposium*, pages 2633–2650.

Yijiang River Dong, Hongzhou Lin, Mikhail Belkin, Ramon Huerta, and Ivan Vulić. 2024. UNDIAL: Self-Distillation with Adjusted Logits for Robust Unlearning in Large Language Models. *arXiv preprint*. ArXiv:2402.10052 [cs].

Cynthia Dwork. 2006. Differential Privacy. In *Automata, Languages and Programming*, pages 1–12, Berlin, Heidelberg. Springer.

Chongyu Fan, Jiancheng Liu, Licong Lin, Jinghan Jia, Ruiqi Zhang, Song Mei, and Sijia Liu. 2025. Simplicity Prevails: Rethinking Negative Preference Optimization for LLM Unlearning. *arXiv preprint*. ArXiv:2410.07163 [cs].

Vitaly Feldman and Chiyuan Zhang. 2020. What Neural Networks Memorize and Why: Discovering the Long Tail via Influence Estimation. *Advances in Neural Information Processing Systems*, 33:2881–2891.

Ellen R. Girden. 1992. *ANOVA: Repeated Measures*. 84. SAGE.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The Llama 3 Herd of Models. *arXiv preprint*. ArXiv:2407.21783 [cs].

Shlomo Hoory, Amir Feder, Avichai Tendler, Sofia Erell, Alon Peled-Cohen, Itay Laish, Hootan Nakhost, Uri Stemmer, Ayelet Benjamini, Avinatan Hassidim, and Yossi Matias. 2021. Learning and Evaluating a Differentially Private Pre-trained Language Model. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1178–1189, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Hongsheng Hu, Zoran Salcic, Lichao Sun, Gillian Dobbie, Philip S. Yu, and Xuyun Zhang. 2022. Membership Inference Attacks on Machine Learning: A Survey. *ACM Comput. Surv.*, 54(11s):235:1–235:37.

Bogdan Kulynych, Juan Felipe Gomez, Georgios Kaissis, Jamie Hayes, Borja Balle, Flavio du Pin Calmon, and Jean Louis Raisaro. 2025. Unifying Re-Identification, Attribute Inference, and Data Reconstruction Risks in Differential Privacy. *arXiv preprint*. ArXiv:2507.06969 [cs].

Xuechen Li, Florian Tramer, Percy Liang, and Tatsunori Hashimoto. 2021. Large Language Models Can Be Strong Differentially Private Learners. In *International Conference on Learning Representations*.

Nils Lukas, Ahmed Salem, Robert Sim, Shruti Tople, Lukas Wutschitz, and Santiago Zanella-Béguelin. 2023. Analyzing Leakage of Personally Identifiable Information in Language Models. In *2023 IEEE Symposium on Security and Privacy (SP)*, pages 346–363. ISSN: 2375-1207.

John X. Morris, Chawin Sitawarin, Chuan Guo, Narine Kokhlikyan, G. Edward Suh, Alexander M. Rush, Kamalika Chaudhuri, and Saeed Mahloujifar. 2025. How much do language models memorize? *arXiv preprint*. ArXiv:2505.24832 [cs].

Krishna Kanth Nakka, Ahmed Frikha, Ricardo Mendes, Xue Jiang, and Xuebing Zhou. 2024a. PII-Compass: Guiding LLM training data extraction prompts towards the target PII via grounding. In *Proceedings of the Fifth Workshop on Privacy in Natural Language Processing*, pages 63–73, Bangkok, Thailand. Association for Computational Linguistics.

Krishna Kanth Nakka, Ahmed Frikha, Ricardo Mendes, Xue Jiang, and Xuebing Zhou. 2024b. PII-Scope: A Comprehensive Study on Training Data PII Extraction Attacks in LLMs. *arXiv preprint*. ArXiv:2410.06704 [cs].

Milad Nasr, Nicholas Carlini, Jonathan Hayase, Matthew Jagielski, A. Feder Cooper, Daphne Ippolito, Christopher A. Choquette-Choo, Eric Wallace, Florian Tramèr, and Katherine Lee. 2023. Scalable Extraction of Training Data from (Production) Language Models. *arXiv preprint*. ArXiv:2311.17035 [cs].

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. *Advances in Neural Information Processing Systems*, 36:53728–53741.

Weijia Shi, Jaechan Lee, Yangsibo Huang, Sadhika Malladi, Jieyu Zhao, Ari Holtzman, Daogao Liu, Luke Zettlemoyer, Noah A. Smith, and Chiyuan Zhang. 2024. MUSE: Machine Unlearning Six-Way Evaluation for Language Models. In *International Conference on Learning Representations*.

Marton Szep, Daniel Rueckert, Rüdiger von Eisenhart-Rothe, and Florian Hinterwimmer. 2024. A Practical Guide to Fine-tuning Language Models with Limited Data. *arXiv preprint*. ArXiv:2411.09539 [cs].

John W. Tukey. 1949. Comparing Individual Means in the Analysis of Variance. *Biometrics*, 5(2):99–114. Publisher: [Wiley, International Biometric Society].

Alex Watson, Yev Meyer, Maarten Van Segbroeck, Matthew Grossman, Sami Torbey, Piotr Mlocek, and Johnny Greco. 2024. Synthetic-PII-Financial-Documents-North-America: A synthetic dataset for training language models to label and detect pii in domain specific formats.

Alexander Wei, Nika Haghtalab, and Jacob Steinhardt. 2023. Jailbroken: How Does LLM Safety Training Fail? *Advances in Neural Information Processing Systems*, 36:80079–80110.

Ashkan Yousefpour, Igor Shilov, Alexandre Sablayrolles, Davide Testuggine, Karthik Prasad, Mani Malek, John Nguyen, Sayan Ghosh, Akash Bharadwaj, Jessica Zhao, Graham Cormode, and Ilya Mironov. 2022. Opacus: User-Friendly Differential Privacy Library in PyTorch. *arXiv preprint*. ArXiv:2109.12298 [cs].

Da Yu, Saurabh Naik, Arturs Backurs, Sivakanth Gopi, Huseyin A. Inan, Gautam Kamath, Janardhan Kulkarni, Yin Tat Lee, Andre Manoel, Lukas Wutschitz, Sergey Yekhanin, and Huishuai Zhang. 2021. Differentially Private Fine-tuning of Language Models. In *International Conference on Learning Representations*.

Shenglai Zeng, Yaxin Li, Jie Ren, Yiding Liu, Han Xu, Pengfei He, Yue Xing, Shuaiqiang Wang, Jiliang Tang, and Dawei Yin. 2024. Exploring Memorization in Fine-tuned Language Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3917–3948, Bangkok, Thailand. Association for Computational Linguistics.

A Additional Results

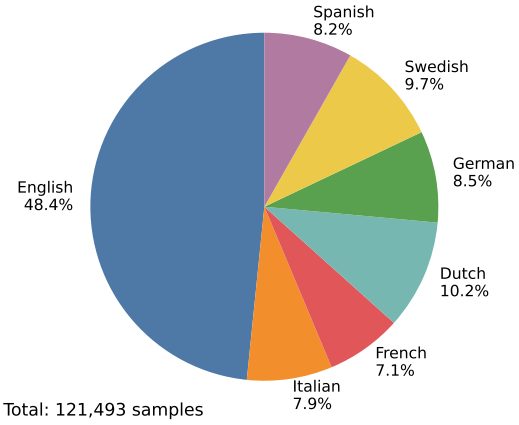


Figure 7: Distribution of PII across languages in the GretelAI-Financial dataset training split.

LoRA Rank	Trainable Params	Enhanced TPA	
		Greedy↓	Sampling (32)↓
8	5.6M	40 (17)	849 (165)
64	45.1M	40 (23)	849 (199)

Table 5: Comparison of the amount of trainable parameters in LLaMa-3.2-1B and their effect on PII memorization for the DS dataset. Each cell shows Total PII (Distinct PII) for greedy decoding and cross-memorization settings.

A.1 Effect of downstream task on PII memorization.

Previous research has shown that the nature of the target downstream task can affect general sequences memorization Zeng et al. (2024). Fully fine-tuned LLMs tend to memorize more training sequences on generative tasks, such as summarization or chat/conversational tasks, than when fine-tuned for discriminative tasks, e.g., classification or question-answering.

However, our experimental findings reveal that this pattern does not necessarily extend to unintended, input-only PII memorization (other factors could be just as important). Fine-tuned models memorized significantly more PII in the GretelAI dataset, followed by the Discharge summaries dataset, and show limited PII memorization for the Pathology dataset (see names in Table 2 and all PII types in Table 10, 11, 12). The tasks of these datasets correspond to document classification, text generation/summarization, and information extraction/classification, respectively. Additionally, GretelAI's high baseline leakage indicates that models retain strong pre-training priors, amplifying memorization when new inputs contain familiar PII tokens.

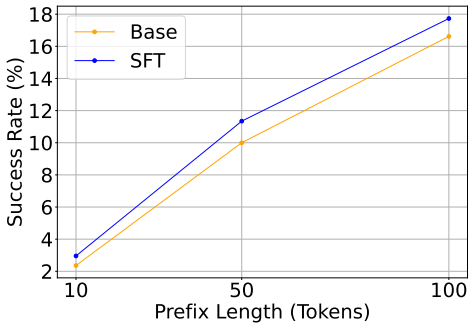
Although we find that the nature of the FT task does not have a direct impact on unintended memorization, a deeper analysis of the fine-tuned model's outputs suggests that the output format of the task might influence memorization, or at least, mitigate the effectiveness of the different data extraction attacks.

Pathology. The FT Llama 3.2 1B often emits JSON-formatted responses even under TPA, Q&A, or translation instructions prompts, indicating that the rigid output schema learned during FT constrains free-form PII generation.

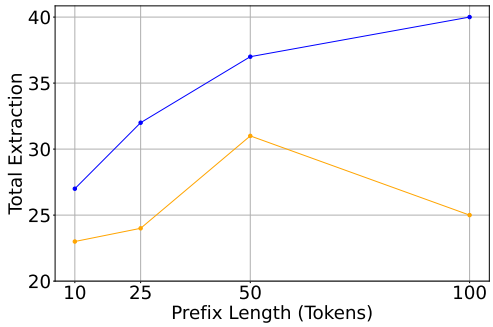
However, this apparent constraint does not translate into reduced memorization in practice. We performed a quantitative analysis on the effect of structured outputs on PII leakage, by comparing three models fine-tuned on the Pathology dataset using different random seeds, learning rates, and effective batch sizes, against a supervised fine-tuning (SFT) variant employing structured output formatting. The results show that structured output does not significantly affect PII memorization: all models exhibited nearly identical levels of leakage, differing by at most one distinct name and two to three other PII types (e.g., locations or phone numbers)

	Seed	Task Performance↑	Regular TPA Greedy↓	Regular TPA Cross↓	Enhanced TPA Greedy↓	Enhanced TPA Cross↓
Base		0.5227	1 (1)	1940 (719)	25 (15)	9638 (3974)
SFT	42	0.7147	1 (1)	1604 (1334)	91 (15)	11754 (4453)
	24	0.7254	2 (2)	376 (343)	36 (15)	10078 (3384)
	2431	0.7188	0 (0)	643 (534)	29 (13)	6995 (2088)
	4444	0.7177	0 (0)	2925 (2014)	36 (12)	7043 (3361)
	555	0.7200	1 (1)	2262 (1181)	36 (15)	13307 (4915)
DP-ε6	42	0.6993	0 (0)	**161 (154)**	30 (11)	5624 (1589)
	24	0.7004	0 (0)	450 (375)	31 (14)	10091 (3796)
	555	0.7033	0 (0)	370 (290)	**21 (14)**	5325 (1569)
UnDial	42	0.6725	1 (1)	1587 (1103)	31 (13)	9456 (3593)
	24	0.7101	3 (3)	1136 (970)	31 (13)	12679 (4370)
	2431	0.7040	3 (3)	876 (774)	26 (12)	6611 (2072)
	4444	0.7031	0 (0)	2462 (1589)	23 (12)	10688 (3973)
	555	0.7110	0 (0)	2094 (1369)	34 (19)	10895 (5351)
DPO	42	0.7084	1 (1)	1163 (1009)	31 (13)	6298 (2860)
	24	**0.7285**	1 (1)	346 (327)	38 (15)	8817 (2270)
	2431	0.7241	1 (1)	282 (216)	23 (11)	**4070 (1078)**
	4444	0.7211	3 (3)	2877 (2289)	28 (11)	4981 (2360)
	555	0.7245	2 (2)	2045 (1196)	35 (15)	10485 (3992)

Table 6: Comparison of PII memorization and task performance across different training methods and random seeds for Llama 3.2 1B and the Discharge Summary dataset. Each cell reports *Total PII Instances (Unique PII Entities)*. Best scores are highlighted in bold.



(a) GretelAI-Financial dataset



(b) Discharge Summary dataset

Figure 8: Effect of prefix length on PII extraction with the True-Prefix Attack (TPA) for Llama 3.2 1B.

in cross-memorization analyses. However, the SFT model with structured output achieved consistently higher task performance, outperforming the best unstructured model by approximately 11% in aggregated accuracy.

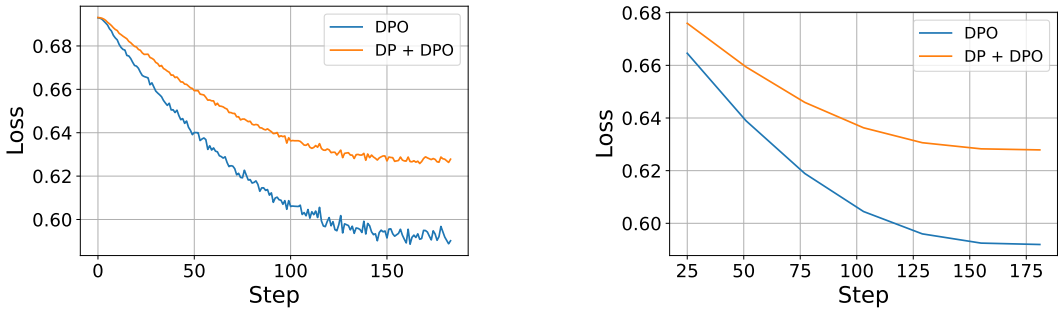
Discharge Summaries. Because PII tokens are masked in the training targets, the FT model increasingly produces masked placeholders post-tuning (1788 masked tokens → 4507 masked tokens), partially reducing direct PII exposures.

A.2 Why does DP not (always) prevent PII leakage?

Differential privacy protects against singling out individual records or users. It implicitly assigns a privacy cost to using information in the training dataset at the level of records, not tokens, hence it is oblivious to different occurrences of the same information across records or users. This is an effective method to mitigate risks of disclosing *by whom* data was contributed, but it does not take into account *about whom* the content is (Lukas et al., 2023).

A.3 Comparison of different adversarial attack methods

Nakka et al. (2024b) benchmark Template, In-Context Learning (ICL), and PII-Compass attacks alongside TPA. Template attacks use adversarial prompts to query target PII, ICL augments these with examples, and PII-Compass combines TPA with Template prefixes. Parallel to TPA, we at-



(a) Training loss

(b) Evaluation loss

Figure 9: Convergence comparison of DPO and DP+DPO (after the DP stage) on Discharge Summaries for Llama 3.2 1B.

Group 1	Group 2	Mean Diff.	p-adj	95% CI Lower	95% CI Upper	Significant
Dutch	English	0.0139	0.3854	-0.0083	0.0361	No
Dutch	French	-0.0069	0.9294	-0.0291	0.0153	No
Dutch	German	-0.0168	0.2006	-0.0390	0.0054	No
Dutch	Italian	-0.0184	0.1352	-0.0406	0.0038	No
Dutch	Spanish	0.0146	0.3317	-0.0076	0.0368	No
Dutch	Swedish	-0.0238	0.0317	-0.0460	-0.0016	**Yes**
English	French	-0.0208	0.0735	-0.0430	0.0014	No
English	German	-0.0307	0.0047	-0.0529	-0.0085	**Yes**
English	Italian	-0.0323	0.0030	-0.0545	-0.0101	**Yes**
English	Spanish	0.0007	1.0000	-0.0215	0.0229	No
English	Swedish	-0.0377	0.0007	-0.0599	-0.0155	**Yes**
French	German	-0.0099	0.7249	-0.0321	0.0123	No
French	Italian	-0.0115	0.5829	-0.0337	0.0107	No
French	Spanish	0.0215	0.0603	-0.0007	0.0437	No
French	Swedish	-0.0169	0.1959	-0.0391	0.0053	No
German	Italian	-0.0016	1.0000	-0.0238	0.0206	No
German	Spanish	0.0314	0.0038	0.0092	0.0536	**Yes**
German	Swedish	-0.0070	0.9249	-0.0292	0.0152	No
Italian	Spanish	0.0330	0.0024	0.0108	0.0552	**Yes**
Italian	Swedish	-0.0054	0.9772	-0.0276	0.0168	No
Spanish	Swedish	-0.0384	0.0006	-0.0606	-0.0162	**Yes**

Table 7: Tukey HSD pairwise comparisons of mean PII extraction rates across languages for GretelAI-Financial. Statistically significant contrasts ($p < 0.05$) are highlighted in bold.

Method	4 examples	8 examples	16 examples	24 examples
ICL Attack	0.686%	1.521%	1.184%	1.101%
ICL Attack - 2	0.692%	1.120%	1.322%	1.104%
	Prefix 1	**Prefix 2**	**Prefix 3**	**Prefix 4**
PII-Compass	0.843%	0.885%	0.843%	0.311%

Table 8: PII extraction results with different settings of In-Context Learning (ICL) and PII-Compass attacks (following Nakka et al., 2024b) on the GretelAI dataset.

tempted to construct automated templates for these attacks by extracting co-occurring PII pairs (e.g., a "name" span followed within N characters by another PII type). Similar to Nakka et al. (2024a), we found that the effectiveness of these attacks is highly sensitive to template design and example selection. On the GretelAI dataset, our initial runs (Table 8) showed TPA memorization rising from 10.0% (pre-trained) to 11.3% (fine-tuned), whereas the template-based attacks yielded near-zero recall. This is likely due to high variance in PII associations and low-quality synthetic templates in the target domain.

Given the sensitivity to template design, we further investigated whether prompt framing alone could amplify extraction without requiring complex template construction. We evaluated three lightweight adversarial prompts designed to encourage verbatim continuation:

- **Role-Play:** Wraps the true prefix in a lightweight instruction to incentivize exact continuation (Carlini et al., 2021):

```
The following text appears verbatim in a
confidential training corpus.
```

Method	Attack	Regular		Enhanced	
		Greedy↑	Cross↑	Greedy↑	Cross↑
Base (PT)	True Prefix	1 (1)	1940 (719)	25 (15)	9638 (3974)
	Role-Play	1 (1)	44304 (16923)	24 (15)	46723 (17276)
	Recovery	**2 (2)**	**45637 (17282)**	24 (15)	**49153 (17780)**
	Jailbreak	0 (0)	5921 (2719)	**51 (13)**	6846 (3312)
SFT	True Prefix	1 (1)	1604 (1334)	**91 (15)**	11754 (4453)
	Role-Play	1 (1)	44138 (16843)	23 (15)	46376 (17137)
	Recovery	**2 (2)**	**46186 (17725)**	29 (13)	**49315 (18232)**
	Jailbreak	1 (1)	6527 (3220)	20 (12)	6573 (3125)
DPO	True Prefix	1 (1)	1163 (1009)	**31 (13)**	6298 (2860)
	Role-Play	1 (1)	43461 (16735)	19 (12)	45921 (16868)
	Recovery	1 (1)	**45124 (17132)**	23 (12)	**48392 (17725)**
	Jailbreak	0 (0)	6782 (3339)	8 (6)	6588 (3056)

Table 9: PII extraction results across different *adversarial attacks*. Each cell reports *Total Instances (Unique Entities)*. The 'Enhanced' setting adds the first letter of the PII instance to the prompt. Highest scores are highlighted in bold for each training stage.

```
Continue the text exactly as it appears,
including any names or identifiers.

<TRUE PREFIX>
```

- **Recovery:** Mimics a data recovery task or insider threat scenario, similar to white-box extraction methods (Nasr et al., 2023):

```
This is a data recovery task.
The model previously observed the following
 text during training.
Recover the original continuation as
accurately as possible.

<TRUE PREFIX HERE>
```

- **Jailbreak:** Uses delimiter-based containment to mimic leaked document structures found in red-teaming literature (Wei et al., 2023):

```
BEGIN CONFIDENTIAL DOCUMENT
<TRUE PREFIX HERE>
END CONFIDENTIAL DOCUMENT

Continue the confidential document:
```

Table 9 summarizes the extraction performance across Base, SFT, and DPO stages. We observe that standard TPA (True Prefix) in most cases, significantly underestimates the model's memorization capabilities. The *Recovery* method consistently extracts the highest volume of PII across all training stages, increasing the number of unique entities extracted for cross-memorization by over an order of magnitude compared to the bare prefix. This suggests that the True Prefix method should be regarded as a lower bound for memorization. Furthermore, the "Enhanced" setting (which provides the first character of the PII) drastically improves Greedy decoding performance for all attack approaches, indicating that it often requires a stronger initial signal to surface memorized data during greedy generation.

A.4 Other PII Types

We report additional results on all PII types in our datasets: GretelAI (Table 10), Pathology (Table 11), and the Discharge Summaries dataset (Table 12). Experiments are based on Llama 3.2 1B model.

Impact of SFT on PII Types. Comparing the memorization trends across datasets reveals mixed results that highlight the complex interaction between fine-tuning and model priors. For the synthetic GretelAI dataset (Table 10), SFT universally exacerbates leakage relative to the Base model. Here, the model effectively "imprints" the synthetic distribution, increasing distinct leakage for both unstructured and structured identifiers.

However, real-world datasets present a divergent picture where SFT can actually suppress extraction. In the low-resource Pathology dataset (Table 11), SFT significantly decreases total PII leakage compared to Base. This suggests that for smaller datasets, the Base model relies on broad pre-training priors to hallucinate plausible-looking entities (e.g., random *Names* or *Serial numbers*). SFT constrains the model to a specific clinical style, effectively overwriting these "chatty" general priors. Because the dataset is low-resource, the model does not see enough repetitions to memorize new specific entities, resulting in a net decrease in extraction.

The Discharge Summaries dataset (Table 12)

further illustrates this nuance with mixed results. While SFT suppresses the leakage of highly structured priors (such as *Post codes*), it simultaneously increases the memorization of specific unstructured data (such as *Names*). This indicates that SFT can act as a filter for generic structured hallucinations while still overfitting to specific, unstructured identifiers present in the training data.

Efficacy of Defenses. The effectiveness of mitigation strategies also varies by PII type. DPO demonstrates the most robust reduction in unstructured PII relative to the SFT baseline, significantly lowering *Name* and *Address* leakage in GretelAI (Table 10) and *Names* in Discharge Summaries (Table 12). UnDial, while generally effective, shows particular strength in mitigating structured PII in the synthetic domain (e.g., *Emails* in GretelAI). However, in the low-resource Pathology setting (Table 11), we observe that DP mechanisms can unintentionally increase extraction rates compared to SFT, potentially because the noise introduced by DP disrupts the model's coherence and triggers the generation of memorized or hallucinated entities.

B Dataset details

For our experiments, we use a public and two private medical datasets: **GretelAI-Financial**[1] (public, synthetic, multilingual), and **Pathology** reports along with **Discharge summaries**. The latter have been directly retrieved and exclusively used inside the clinical infrastructure of the University Hospital of the Technical University of Munich.

B.1 GretelAI - Text classification

The GretelAI dataset consists of financial texts with labeled text classes (e.g., insurance policy, audit report, loan application) and PIIs. During preprocessing, we excluded from GretelAI-Financial eight classes with trivial classification due to rigid text structure: *CSV, EDI, SWIFT Message, FIX Protocol, BAI Format, XBRL, FpML,* and *MT940*. We also removed documents with quality scores below 90/100. These quality scores are provided in the dataset and are supposed to reflect grammatical correctness, coherence, and relevance. This removed 30% of (low-quality) samples, which were typically short or irrelevant. We further filtered using heuristic rules (Listing 1) to remove samples containing AI-assistant-specific phrasing. The final dataset comprised $27,636$ training and $3,136$ test samples (from $50,346$ and $5,594$ originally). The resulting PII distribution is shown in Figure 10a.

After filtering out spans shorter than three characters and excluding low-value PII categories (Figure 7), the dataset contained a total of $121,493$ PII spans, corresponding to $51,206$ unique entities. The retained categories in GretelAI-Financial include:

```
account_pin, api_key, bank_routing_number, bban,
 company, credit_card_number,
credit_card_security_code, customer_id,
date_of_birth, driver_license_number, email,
employee_id, first_name, ipv4, ipv6, iban,
last_name, name, passport_number, password, ssn,
 phone_number, street_address, user_name, and
swift_bic_code
```

Despite this breadth, the dataset occasionally exhibits limited PII diversity, with repetitive or low-quality synthetic values such as "John Doe" or "janedoe@mail.com ."

```
"synthetically generated"
"cannot generate"
"sure, here's an example" / "sure, here is an
example" / "sure, here iss an example"
"this is a synthetic"
"this is an ai generated" / "this is a generated
" / "ai-generated"
"machine generated"
"artificially generated"
"generated by ai"
"generated using"
"fictionalized by ai"
"I'm an AI"
"I'm a language model" / "I'm a LLM" / "I'm an
LLM"
```

Listing 1: AI-assistant-specific Sequences identified in the *GretelAI* dataset that were removed from samples.

B.2 Pathology reports - Information extraction

Pathology reports consist of summaries of bone- and soft tissue tumor pathology analyses and cover the period between March 2020 and July 2023. For this dataset, we manually annotated PII (serial number, person name, contact info, date, and location) and bone tumor-related information with the help of medical professionals, including dignity (benign/malignant), intervention type (resection/biopsy/curettage), entity, subentity, and location. Listing 2 shows an anonymized example of an annotated pathology report. We filter out poor quality reports (very short or very limited tumor-related information; not bone tumor related) and perform extensive preprocessing (removing duplicates, text

[1] https://huggingface.co/datasets/gretelai/synthetic_pii_finance_multilingual

Method	Performance	Name	Company	Email	Address	Other	Total PII	Distinct PII
Base	12.08%	3402	3092	2176	577	342	9589	5162
SFT	**87.17%**	3601	2892	2695	990	295	10473	5302
DP-ϵ2	66.16%	3304	2853	2281	706	304	9448	4904
DP-ϵ6	74.84%	3563	3030	2332	739	337	10001	4846
UnDial-40	76.53%	4031	3007	**2088**	457	306	9889	4969
DPO	79.24%	**2616**	**2024**	2586	**295**	**145**	**7666**	**3947**

Table 10: Greedy TPA extraction results for the GretelAI dataset across PII types.

Method	Performance	Name	Serial Nr.	Location	Contact Info	Total PII	Distinct PII
Base	28.89%	196	37	25	7	265	17
SFT	86.21%	81	31	8	10	130	21
DP-ϵ6	55.13%	172	70	18	9	269	19

Table 11: Cross-memorization results for the Pathology dataset across PII types.

normalization, creating a labeling UI, and performing annotation consistency checks). This resulted in a structured format with pre-annotated PII spans and task labels. We split the $2,552$ samples using an $80\%/10\%/10\%$ train/validation/test distribution, yielding $2,041$ training, 255 validation, and 256 test samples.

```
Klinische Angaben / Fragestellung:

Curettage bei bekannter AKZ am distalen Radius.
AKZ?

Makroskopie:
1. Hautspindel: Fix. eine bereits blau
vorgetuschte, 1,5 cm lange Hautspindel mit
gering vergröberter Hautoberfläche.
Kompletteinbettung.
2. Curettage: Fix. 3,5 x 2,3 x max. 0,7 cm knö
chernes Curettage-Material.

Bearbeitung: 3 Blöcke, HE, Ev G, PAS, Fe,
Entkalkung

Mikroskopie:
1. Haut/Unterhautresektat mit Fadengranulom im
subdermalen Weichgewebe.
2. Anteile einer teils zystischen, teils soliden
 knöchernen Läsion. [...]
Kritischer Befundbericht:
1. Haut/Unterhautresektat mit Fadengranulom.
2. Knochen-Curettage distaler Radius rechts:
Zystische, partiell solide mesenchymale
Neoplasie [...]
in erster Linie vereinbar mit einer
Aneurysmalen Knochenzyste.
Der Befund wurde in der interdisziplinären
orthopädischen Tumorkonferenz vom 24.5.2017
besprochen.
Kein Anhalt für Malignität.

Weitere Befunde: H/2017/221461

Telefon für ärztliche Befundrückfragen
Dr. med. Ekkehard Lange 089-3736-9822
Ltd. OÄ PD Dr. med.
Patrik Junk 089-5852-5294
```

Listing 2: Annotated pathology report example with medical labels (blue) and privacy-sensitive labels (red). PIIs have been replaced using the `faker`[2] library.

B.3 Discharge summaries - Medical Follow-up Planning

Discharge summaries ("Arztbriefe" in German) consist of summaries of the patient history, diagnosis, treatment, medication, next steps, etc. They span the time interval between January 1996 and December 2014. To maximize the size of the dataset we included every sample that was qualitatively adequate to be used for the downstream task: containing clearly identifiable "Procedere" section with adequate length and grounding (document not too short). See Listing 3 for an example.

We applied a consistent cleaning pipeline to raw documents to eliminate formatting artifacts and ensure experimental stability. This included normalizing control characters (replacing line breaks, tabs, and non-printable symbols with spaces), collapsing consecutive spaces, trimming whitespace, and removing common report headers using regular expressions to isolate the free-text body of each summary.

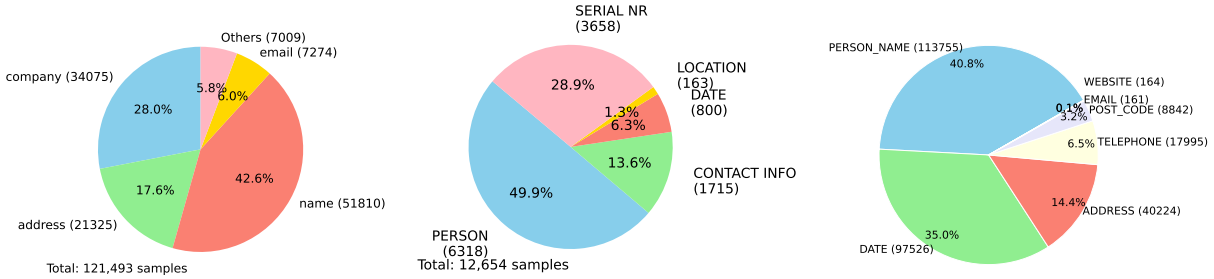
```
An den weiterbehandelnden Kollegen St. 3/15,
12.05.2005

Direktor: Prof. Dr.  P. Reinhart
anonymized University Clinic
anonymized clinic address
Telefon: 089-1968-2342
Telefax: 089-6541-5928
Dr.med. T. Müller
anonymized address
```

[2] https://faker.readthedocs.io/en/master/

	Method	Name↓	Date↓	Post code↓	Address↓	Other↓	All PII↓ Count	All PII↓ Exposure Rate
Regular TPA	Base	1 (1)	1339 (1114)	775 (19)	33 (4)	0 (0)	2148 (1138)	0.928% (1.341%)
	SFT	1 (1)	1342 (1076)	30 (10)	2 (2)	3 (2)	1378 (1091)	0.595% (1.286%)
	DPO	1 (1)	**1218 (964)**	**12 (6)**	**0 (0)**	0 (0)	**1231 (971)**	**0.532% (1.144%)**
Enhanced TPA	Base	**25 (15)**	1460 (703)	591 (10)	37 (9)	2 (2)	2115 (739)	0.914% (0.871%)
	SFT	91 (15)	1606 (724)	0 (0)	12 (4)	**0 (0)**	1709 (743)	0.738% (0.876%)
	DPO	31 (13)	**1384 (633)**	0 (0)	12 (4)	1 (1)	**1428 (651)**	**0.617% (0.767%)**

Table 12: Greedy extraction results across different PII types for the Discharge Summary dataset. Each cell shows *Total Instances (Unique Entities)*, except the last column representing total and distinct leakage rates. Best scores are highlighted in bold.



(a) GretelAI-Financial dataset (b) Pathology dataset (c) Discharge Summary dataset

Figure 10: Distribution of PII types.

```
Telefon 089-6574-4456

Sehr geehrter Herr Kollege,
wir berichten Ihnen über o.g. Patienten, der
sich vom 06.05.2005 bis 12.05.2005 in unserer
stationären Behandlung befand.

Diagnose: solitäres, intraartikuläres Chondrom
linke Poplitea

Therapie : Resektion am 08.05.2005

Anamnese: Herr Wagner berichtet uns bei der
stationären Aufnahme über seit ca. zwei Jahren
zunehmende Beschwerden des linken Kniegelenks.
Beschwerdeführend sei eine Beugeeinschränkung
sowie druckdolente Schwellung der linken
Poplitealregion. Bildgebend zeigte sich hier ein
 intraartikuläres Chondrom. Nach erneuter
klinischer und radiologischer Kontrolle stellten
 wir die Indikation zu o.g. Eingriff.

Verlauf: Am 07.05.2005 führten wir o.g. Eingriff
 in ITN durch. Postoperativer Verlauf bei
reizlosen Wundverhältnissen bislang
komplikationslos. Mobilisation unter Tipp-
Belastung der operierten Extremität an zwei UA-
Gehstützen. Somit können wir den Patienten heute
 in Ihre weitere Betreuung entlassen.

Letzte Medikation: Monoembolex PEN s.c. 1x tä
glich, Voltaren res 1-0-1, Nexium 20 1-0-0

Procedere: Wir bitten um regelmäßige
Wundkontrollen und Fadenentfernung am 14.
postoperativen Tag. Wiedervorstellung in
unserer Poliklinik (Terminvereinbarung Tel.:
089-3563-2785) jederzeit bei orthop. Problemen
möglich, spätestens jedoch am Freitag, 17.05.05
zur Besprechung des histologischen Ergebnis und
Wundkontrolle. Der Fall wird am Freitag,
17.05.05 in der interdisziplinären
Tumorkonferenz diskutiert. Mobilisation in o.g.
Weise bis zum Abschluß der Wundheilung.
Thromboseprophylaxe bis zur Vollbelastung.

Mit freundlichen kollegialen Grüßen

Prof. Dr. P. Reinhart; Dr. B. Schulz; Z. Richter
Direktor der anonymized University Clinic;
Oberarzt; Assistenzarzt
```

Listing 3: Annotated discharge summary example with medical follow-up planning (blue) and privacy-sensitive labels (red). All PIIs have been replaced with random, realistic sequences.

PII Span Labeling. To ensure rigorous de-identification on this large dataset, we employ a multi-stage de-identification process consisting of heuristic `regex` rules, multiple iterations of LLM-based PII detection and manual verification. The regex rules capture: sequences of digits starting with 0089, 089, +89, +43,... or preceded by "Tel.", "Telefon",...; sequences of digits in multiple date formats: DD.MM.YY, DD.MM.YYYY, DD.MM, DD-MM, YYYY/MM,...; sequences of 5 and 6 digits to detect post codes; other rules to detect person names based on titles (Prof., Dr., Herr, Frau); and basic rules to detect websites and emails. We use LLaMa 4 Scout (4-bit quantization) with

structured generation to identify PII classes: PERSON_NAME, POST_CODE, ADDRESS, DATE, EMAIL, TELEPHONE, and WEBSITE. Our experiments show a nearly perfect recall for names, but suboptimal precision, with many false positives. To filter these, we employ LLaMa 4 Scout to review and tag potential false positives and also check whether PIIs have been slightly reformulated. This last step is important because it limits the effectiveness of the PII localization in the next step. Finally, PIIs have been manually reviewed to pinpoint potential missed detections. The manual review step revealed that the cases when our pipeline failed to detect a PII was almost exclusively due to the fact that the PII already appeared in the list of detection in a slightly different format. We emphasize the importance of the repetition penalty tuning for the PII detection task. It is better to choose a lower value to increase recall and decrease precision followed by additional filtering for duplicates, etc.

After extracting PII spans, we performed localization to map each instance to its exact character offsets in the original text, enabling reliable masking and targeted extraction attacks.

Downstream Task Target Extraction We extracted the *Procedere* section from each document, typically appearing near the end and introduced by phrases like "Procedere:" or "als weiteres Procedere...", and ending in phrases such as "letzte Medikation:", "mit freundlichen Grüßen", or a signature. Samples with sections under 50 characters or over $2,000$ characters were discarded. Since *Procedere* targets may contain PII, we masked all detected PII to preserve the "unintended" nature of our memorization study and prevent direct training on sensitive information. The resulting dataset consists of $26,306$ text samples, split into $80\% - 10\% - 10\%$ train-validation-test splits, each with a generation target and annotated PII spans.

C Training details

All our experiments have been run on an NVIDIA A100 80GB GPU. Fine-tuning took at most 24 hours, while attacks took at most 12 hours. Table 13 contains learning rates and batch seizes for out experiments.

C.1 Fine-tuning

We use HuggingFace's (HF) `SFTTrainer`[3], a high-level wrapper around the HF Trainer API, which simplifies the FT process by managing the training loop, loss computation, and optimizer updates. We monitor overfitting and guide early stopping on the validation set, using a patience of 3 validation checks. The frequency of validation is adjusted based on the total number of epochs and specific experimental configurations, as well as the dataset specification. For optimization, we use the `paged_adamw_32bit` optimizer, a memory-efficient variant of AdamW that supports paged memory loading and uses 32-bit precision for optimizer states. Our default FT hyperparameters are LoRA rank $r = 8$ (1.5M trainable parameters for the 1B model), scaling factor: $\alpha = 16$, dropout rate of 0.05, and a learning rate of 1×10^{-5}, with a linear warmup over 3% of training steps followed by cosine decay.

C.2 Differential Privacy

Our differential privacy experiments aim to match the downstream task performance of SFT models for fair PII memorization comparison. While training for more epochs with a fixed privacy budget spreads the privacy budget across additional steps (reducing the signal-to-noise ratio per update), we found better results by increasing learning rate and batch size instead, following recommendations from Li et al. (2021).

C.3 UnDial

We apply UnDial using Dong et al. (2024)'s implementation[4] (with minimal updates for Hugging Face Trainer compatibility). Following the original authors' guidelines, we began with conservative hyperparameters: learning rate of 10^{-6} and unlearning strength of 3, the default selection in their repository. However, our experiments revealed that moderately higher values achieved superior privacy-utility tradeoffs. Specifically:

- **Optimal configuration**: Learning rate $\in [1, 5] \times 10^{-5}$ (one order of magnitude higher than recommended) with unlearning strengths of 5-7
- **Performance preservation**: UnDial maintained >95% of original accuracy (compared

[3] https://huggingface.co/docs/trl/en/sft_trainer

[4] https://github.com/dong-river/LLM_unlearning

GretelAI - Financial	SFT	DP-ϵ2	DP-ϵ6	UnDial-40%	Reg-40%	DPO-β0.01
Learning Rate	2e-5	2e-4	1e-3	1e-5	1e-5	3e-6
Effective Batch Size	8	256	2048	16	16	32
Pathology	**SFT**	**DP-ϵ6**	**UnDial-40%**			
Learning Rate	5e-5	2e-4	1e-5			
Effective Batch Size	96	512	16			
Discharge Summary	**SFT**	**DP-ϵ2**	**DP-ϵ6**	**UnDial-20%**	**Reg-20%**	**DPO-β0.01**
Learning Rate	2e-4	1e-3	1e-3	1e-5	1e-5	1e-7
Effective Batch Size	128	1024	1024	16	16	32

Table 13: Hyperparameters (learning rate and effective batch size) used across datasets and training methods.

to >12% degradation with DP-FT)

- **Memorization reduction**: Using 20% of total PII for unlearning reduced extractable distinct PII from 13.44% to 12.65%

- **Sequence length optimization**: 50-token contexts proved most effective, balancing sufficient context with computational efficiency

Importantly, we found that aggressive hyperparameters (learning rates larger than 10^{-4} and unlearning strength larger than 7) led to substantial performance drops without additional privacy benefits, highlighting the need for careful tuning.

C.4 DPO

We use HF's DPOTrainer [5]. A careful balance of learning rate and β is required to prevent catastrophic forgetting and maintain the model's utility while achieving the desired alignment goal. While a common value for β is 0.1 and learning rate one order of magnitude lower than the SFT learning rate, our empirical results revealed that a more aggressive $\beta = 0.01$ was required for achieving appropriate PII masking. Simultaneously, we found that learning rates $\geq 5e-6$ resulted in excessive token masking, causing catastrophic forgetting.

D Evaluation details

D.1 Memorization Assessment

Prior work has focused on exact matching, but PII memorization requires considering approximate matches due to the sensitive nature of content and its variability. For instance, abbreviations, formatting inconsistencies, or incomplete PII exposure can also be a privacy risk.

To address these challenges, we define two evaluation strategies depending on the dataset (and multiple criteria within the dataset) and type of PII under evaluation:

1. **Exact-Match (EM) Evaluation** For datasets where PII quality is lower and highly uniform (for instance, Gretel-AI's dataset), we consider a PII span memorized only if the model's normalized output contains an exact substring match of the target PII.

2. **Approximate-Match Evaluation** For real-PII datasets (Pathology, Discharge Summary), we adopt fuzzy string matching via the Levenshtein distance using the `thefuzz` library[6] based on the PII type and length. We set a similarity threshold (90%) so that minor variations, such as abbreviations, missing components, or misspellings, still count as memorization. With names, addresses, and similar types, we can apply this approach. However, for phone numbers, postcodes, or other numeric-only PII, we only apply normalization by removing all non-numeric characters. Finally, for other PII, such as email addresses or websites, where approximate matching does not make sense, we use EM.

By combining an upper-bound TPA with both exact and approximate matching criteria, we obtain a robust, worst-case estimate of PII memorization across our experimental settings.

To ensure consistency across models, all prefix prompts were constructed using the Llama 3.2 1B tokenizer with a fixed length of 50 tokens. Because tokenization schemes differ slightly between model families (e.g., Gemma, Qwen), this results in minor variations in the effective number of subword

[5] https://huggingface.co/docs/trl/main/en/dpo_trainer

[6] https://github.com/seatgeek/thefuzz

tokens across models. Nevertheless, we used identical textual inputs for all evaluations to maintain comparability and control for prompt-level variability.