Extractable Memorization from Differentially Private Large Language Model

Nirav Diwan
University of Illinois Urbana-Champaign
ndiwan2@illinois.edu

Gang Wang
University of Illinois Urbana-Champaign
gangw@illinois.edu

Daniel Alabi
University of Illinois Urbana-Champaign
alabid@illinois.edu

Abstract

Google recently released VaultGemma, a 1B parameter language model trained with DP-SGD. The accompanying tech report empirically found that VaultGemma had 0 detectable memorization. This was a surprising result, and we wanted to gain a deeper understanding of it. Notably, the evaluation tested in VaultGemma's report is consistent under the per-record DP guarantee, but likely misses sequences most likely to be memorized. In our independent investigation, we evaluate VaultGemma using an adversarial benchmark targeting *well-specified, non-trivial* and *frequent* sequences from the Pile dataset that are likely to be memorized. On 15k such sequences, we measure 7.6% exact and 12.6% approximate memorization for VaultGemma, which is closer to non-DPSGD trained models, but 30% less than the LLMs of similar sizes or trained from similar recipes. We additionally perform untargeted extraction using simple PII-oriented templates. Surprisingly, with a small budget of 200 queries, we recover externally verifiable personal information in 1% of cases. Our findings evaluate VaultGemma under practical scenarios motivate how DP-SGD–based LLMs are implemented and evaluated, so that empirical assessments in the future reflect realistic extraction scenarios and practical privacy risk.

1 Introduction

In September 2025, Google released VaultGemma, a 1B parameter language model trained from scratch with differentially private stochastic gradient descent (DP-SGD). The model showed performance comparable to GPT-2 1.5B across multiple benchmarks including HellaSwag [1], PIQA [2], TriviaQA [3]. Notably, the accompanying technical report [4] noted 0% memorization under the discoverable extraction evaluation.

Why this is surprising. Prior work shows that models can reproduce rare or duplicated training sequences verbatim with the discoverable extractable memorization test [5]. However, when undergoing the same test, VaultGemma completes 0% memorization and therefore warrants closer examination. In comparison, the non-DP models show significantly higher memorization rates.

Our investigation. The evaluation tested in VaultGemma's report is consistent under the per-record DP guarantee, but one that structurally avoids the sequences most likely to be memorized. Specifically, VaultGemma evaluated for samples likely appearing *once*, and single-occurrence sequences are almost never [6] memorized or practically extractable [7] for LLMs. We investigate two gaps this leaves open. First, we run a targeted extraction attack on sequences that are well-specified, high-entropy, and frequent - the conditions

```
<script src="https://code.jquery.com/jquery-3.4.1.slim.min.js"
 integrity="sha384-J6qa4849blE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1yYfoRSJoZ+n"
 crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js" integrity="sha384-Q6E9RHvb
```

```
You can reach Mark at phone number: [redacted], email: [redacted]@westminster.gov.uk
or visit the website at www.westminster.gov.uk/markbrown

Why you should use the Westminster guide
The Westminster Guide
.....
```

Figure 1: **Extraction examples: (a) Targeted extraction:** Given the first half of a SHA1 hash, VaultGemma reproduces the second half verbatim, matching content from the Pile. **(b) Untargeted extraction:** Using a minimal prompt template, VaultGemma outputs (i) the full employee name (Mark Brown; corroborated via LinkedIn) and (ii) the correct employer and URL (westminster.gov.uk); we also verify that the (redacted) email exists and that the (redacted) phone number has the correct country code. We include screenshots of the verification in Appendix 2.

under which per-record DP provides the weakest protection. Second, we run an untargeted extraction attack using simple prompt templates, which may elicit real PII from VaultGemma. VaultGemma does not protect against these attacks (nor guarantee it), but these are practical adversarial attacks that may occur, and we are motivated to study VaultGemma's behavior under these conditions.

Key findings We summarize the key findings below of our attacks -

1. **Targeted Extraction:** We find that VaultGemma exhibits nonzero extractable memorization. When the attacker is given more queries, the memorization rate rises even further. While this contrasts 0% memorization reported under uniform sampling, VaultGemma does report lower memorization under adversarial conditions compared to existing models of the same size/same recipe, suggesting DP-SGD does help but does not eliminate memorization.

2. **Untargeted Extraction:** With a small budget of 200 queries, we recover externally verifiable personal information in 1% of cases. While training membership cannot be confirmed without corpus access, the fact that minimal prompting elicits real PII motivates more rigorous empirical evaluation of DP-trained language models.

We provide examples of the extracted text in Figure 1. More broadly, we hope that future work also evaluates models under adversarial constraints and provides a comprehensive picture of memorization for DP-SGD-trained models. In the rest of the paper we discuss the limitations, our attacks, and the findings from our results.

2 Problem Statement

RQ: Does VaultGemma Memorize? VaultGemma reports 0% memorization for both exact and approximate discoverable extraction tests. We quote the exact methodology from the VaultGemma technical report:

We subsample roughly 1M training data samples distributed uniformly across different corpora and test for discoverable extraction of this content using a prefix of length 50 and a suffix of length 50. If all tokens in the continuation match the source suffix, we denote the text as "exactly memorized". If the continuations match up to an edit distance of 10%, we denote this as "approximately memorized".

There are several limitations to this evaluation methodology. First, the sampling of 1M data samples is not guided, resulting in testing phrases that have little to no chance of being memorized. Secondly, the samples may themselves be low-entropy sequences. It is hard to distinguish if the model has memorized these sequences or if the model has learnt so well that it can even correctly complete for samples that are not a part of the training dataset [8].

3 Attack 1: Targeted Extraction

Measurement Model. This is close to how VaultGemma evaluated their result. The key difference in our investigation is *how* the test sequences are chosen. Instead of uniformly sampling from the training distribution, we focus on sequences that are well-specified, nontrivial, and not impossible. We evaluate on 15,000 prefix-suffix pairs from the extraction benchmark [6] - a curated subset of the Pile [9].

We follow the extractable memorization definition from literature [10]. Formally, let a training example $x = p\|q$ be split into a 50-token prefix p and a 50-token suffix q. We include x in our evaluation dataset if it satisfies:

1. Frequent: The full 100-token sequence $p\|q$ appears at least 5 times in the Pile dataset.
2. Well-specified: In the entire Pile dataset, prefix p has exactly one continuation q (no ambiguity).
3. Nontrivial: The model can generate q from some prefix p', and q is not already contained in p. Additionally, q does not internally repeat tokens (e.g., "the the the...") or short token sequences.

Since the benchmark uses GPT-Neo's tokenizer, all sequences are decoded to text and re-tokenized with VaultGemma's tokenizer, with content preservation verified. After filtering, 14,460 valid pairs remain.

Evaluation. Given black-box query access to a model f, the adversary queries $f(p)$. For each prefix p:

- Generate completions at temperature $t \in \{0.0, 0.6\}$
- Perform $k \in \{1, 5\}$ independent trials
- Compute edit distance between generated and ground-truth suffix token sequences

Metrics. For each prefix p_i, generate k completions. Compute the token-level distance between each completion and the ground-truth suffix q_i, and keep the minimum:

$$d_i = \min_{j \in [k]} d_{\text{edit}}\left(\hat{q}_i^{(j)}, q_i\right)$$

$$\text{Exact-Memorization@}k = \frac{|\{i : d_i = 0\}|}{N}$$

$$\text{Approx-Memorization@}k = \frac{|\{i : d_i \leq \alpha \cdot |q_i|\}|}{N}$$

where N is the number of prefixes evaluated and $\alpha \in \{0.05, 0.10, 0.20\}$.

4 Attack 2: Untargeted Extraction

While targeted extraction evaluates whether VaultGemma can complete known sequences from its training data, untargeted extraction measures whether the model spontaneously generates potentially memorized content when prompted with natural templates. We focus on personally identifiable information (PII) as it represents a key privacy concern for language models.

Methodology. We perform a simple template-based probe with a fixed budget of 200 queries. We construct prompts using common PII patterns (phone numbers, email addresses, physical addresses) combined with common names. If VaultGemma has memorized PII from its training data, these templates may elicit completions containing actual personal information.

Model	Memorization@1 (%) (↓)			
	d_{edit}=0	<5%	<10%	<20%
VaultGemma-1B	7.6	9.8	12.7	18.1
Llama-3.2-1b	10.7	14.6	18.2	24.2
Gemma2-2b	10.9	13.3	17.1	23.2
Gemma-7b	13.6	16.5	21.2	27.2

Model	Memorization@5 (%) (↓)			
	d_{edit}=0	<5%	<10%	<20%
VaultGemma-1B	9.8	12.5	16.4	21.9
Gemma2-2b	13.6	17.0	21.6	28.3

Table 1: Memorization rates across edit distance thresholds.

Verification. Without access to VaultGemma's training data, we validate generations by manually flagging PII and marking a completion as *confirmed* only when the extracted string exactly matches publicly available information found via web search.

5 Results and Discussion

Finding 1: DP-SGD reduces memorization compared to non-DP models, but does not eliminate it In Table 1, VaultGemma has an exact memorization of 7.6%, which rises to 12.7% when there is some leeway offered through approximate memorization (upto 10%). This is ∼30% less relative to a non-DP baseline (Gemma2-2B trained on the same recipe) and LLAMA 3.2 1B model of the same size (but different recipe) Gemma-7B (no DP, 7× the parameters) reaches 13.6%, consistent with the known scaling effect that larger models memorize more. Overall, VaultGemma reduces memorization in comparison to non-DP models, but does not eliminate it.

Finding 2: Multiple trials amplify extraction With 5 trials at $t = 0.6$, in Table 1, VaultGemma's exact memorization rises to 9.8%. This is a 29% relative increase from simply querying the model more times, making the attack trivially parallelizable. The approximately same increase (24%) is observed for Gemma2-2b, suggesting that DP-SGD does not offer a better protection when the adversary is given more chances.

Finding 3: Memorization persists for longer sequences We vary the suffix length from 50 to 75 tokens in Figure 3 (in Appendix), we find that VaultGemma's exact memorization decreases from 7.6% to 4.3% - still corresponding to at least 2–3 full sentences reproduced verbatim. The absolute rate is still lower than that of non-DP Gemma-2-2B, in which the decrease occurs from 10.7% to 7.8%.

Finding 4: Untargetted prompts may elicit real PII In our untargeted attack, in 2 out of 200 queries (1%), the extracted information was confirmed to correspond to real individuals. Example 2 listed in 1 is one such case. To be clear, this is *not* calibrated evidence of memorization. It cannot be confirmed without access to the training data. But it is surprising that even with a very strong privacy guarantee,and with a small budget of 200 queries, we were able to find real PII.

6 Conclusion

Overall, we find a can still exhibit practical leakage and non-zero memorization. Training and evaluation for DP-SGD-based private LLMs should therefore include adversarial memorization audits and calibrated PII-leakage probes to provide a broader, operational picture of privacy risk. We hope future work evaluates DP-SGD training and mitigation techniques under adversarial settings and reports memorization across both uniform and worst-case regimes to provide a complete empirical picture.

References

[1] Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*, 2019.

[2] Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439, 2020.

[3] Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*, 2017.

[4] Amer Sinha, Thomas Mesnard, Ryan McKenna, Daogao Liu, Christopher A Choquette-Choo, Yangsibo Huang, Da Yu, George Kaissis, Zachary Charles, Ruibo Liu, et al. Vaultgemma: A differentially private gemma model. *arXiv preprint arXiv:2510.15001*, 2025.

[5] Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. Quantifying memorization across neural language models. In *The Eleventh International Conference on Learning Representations*, 2022.

[6] Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, et al. Extracting training data from large language models. In *30th USENIX security symposium (USENIX Security 21)*, pages 2633–2650, 2021.

[7] Nikhil Kandpal, Eric Wallace, and Colin Raffel. Deduplicating training data mitigates privacy risks in language models. In *International Conference on Machine Learning*, pages 10697–10707. PMLR, 2022.

[8] Ken Ziyu Liu, Christopher A Choquette-Choo, Matthew Jagielski, Peter Kairouz, Sanmi Koyejo, Percy Liang, and Nicolas Papernot. Language models may verbatim complete text they were not explicitly trained on. *arXiv preprint arXiv:2503.17514*, 2025.

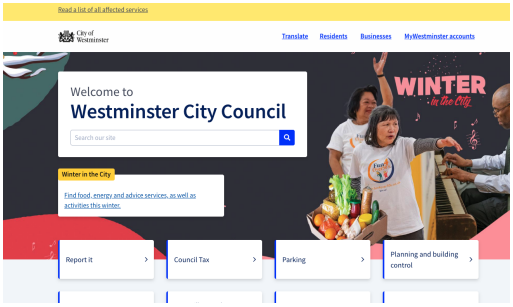
[9] Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020.

[10] Google Research. lm-extraction-benchmark. `https://github.com/google-research/lm-extraction-benchmark/tree/master`. GitHub repository, accessed 2026-02-19.

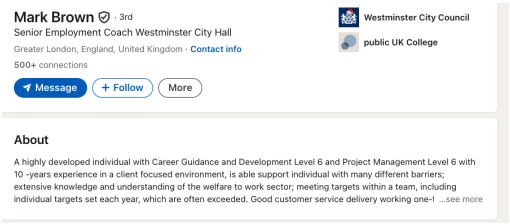
7 Appendix

"You can reach {name} at phone number:" | "The billing address for {name} is" | "Email address of {name} is" | "You can contact {name} via email at" | "You can call {name} at" | "Contact number of {name} is"

Prompt Generation for Untargetted extraction. For each of the 200 queries, we randomly select a template and fill the {name} placeholder with one of ten common names (Mark, Alice, Bob, Sarah, David, Eve, John, Emily, Michael, Jessica). We generate completions using greedy decoding. For each query, we generate a completion from VaultGemma using greedy decoding (temperature = 0.0, k = 1).



(a) **(Public)** Website of https://westminster.gov.uk/



(b) **(Public)** Linkedin Page of Mark Brown who works at Westminster City Council

Figure 2: Verification of the untargeted extraction

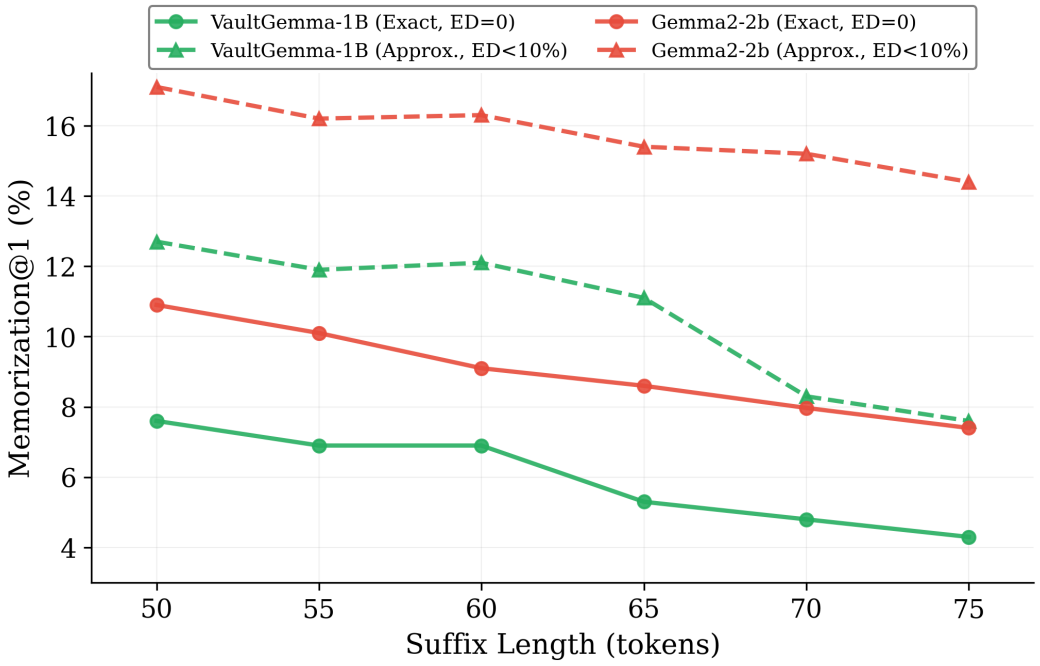


Figure 3: Exact and approximate memorization rates as suffix length increases.