

Privacy Preserving In-Context-Learning Framework for Large Language Models

Bishnu Bhusal¹, Manoj Acharya², Ramneet Kaur², Colin Samplawski²,
Anirban Roy², Adam D. Cobb², Rohit Chadha¹, Susmit Jha²

¹University of Missouri, 411 S 6th St, Columbia, MO 65211, USA

²SRI International, 333 Ravenswood Avenue, Menlo Park, CA 94025, USA

{bhusalb, chadhar}@missouri.edu

{manoj.acharya, ramneet.kaur, colin.samplawski, anirban.roy, adam.cobb, susmit.jha}@sri.com

Abstract

Large language models (LLMs) have significantly transformed natural language understanding and generation, but they raise privacy concerns due to potential exposure of sensitive information. Studies have highlighted the risk of information leakage, where adversaries can extract sensitive information embedded in the prompts. In this work, we introduce a novel private prediction framework for generating high-quality synthetic text with strong privacy guarantees. Our approach leverages the Differential Privacy (DP) framework to ensure worst-case theoretical bounds on information leakage without requiring any fine-tuning of the underlying models. The proposed method performs inference on private records and aggregates the resulting per-token output distributions. This enables the generation of longer and coherent synthetic text while maintaining privacy guarantees. Additionally, we propose a simple blending operation that combines private and public inference to further enhance utility. Empirical evaluations demonstrate that our approach outperforms previous state-of-the-art methods on in-context-learning (ICL) tasks, making it a promising direction for privacy-preserving text generation while maintaining high utility. Our code is available at <https://github.com/bhusalb/privacy-preserving-icl>.

Introduction

Large Language Models (LLMs) have enjoyed widespread success in many applications. Although they primarily obtain foundational knowledge through pre-training, most users tailor trained LLMs through prompt engineering. Compared to the resource-intensive optimization of model parameters during training, prompt engineering is typically performed via API calls, where prompts are progressively refined to achieve optimal downstream performance.

However, this workflow has data privacy risks as sensitive records can be exposed in prompts and responses. For example, when LLMs are deployed with user data, such as clinical reports, incorporated into prompts, there is a risk that sensitive information can be inadvertently disclosed to non-relevant users (Wang et al. 2023). An adversary can also extract sensitive user data in prompts via “jailbreaks”, where even entire prompts or segments of prompts can be extracted

verbatim from some attacks. A mitigation approach could be to scrub Personally Identifiable Information (PII) from prompts, but even with that, there have been cases, such as in linkage attacks, where a combination of secondary information is enough to link the record back to an individual (Powar and Beresford 2023), leading to potential privacy violations.

Differential Privacy (DP) (Dwork, Roth et al. 2014) has been previously used for protecting individual data as it ensures that sensitive information stays confidential while allowing meaningful insights to be drawn from the aggregated dataset. The U.S. Census Bureau’s LEHD OnTheMap tool (Machanavajjhala et al. 2008), Google’s RAPPOR system as part of Google Chrome (Erlingsson, Pihur, and Korołova 2014), and Facebook and Social Science One’s release of election dataset (Messing et al. 2020; Evans and King 2023) are some noteworthy examples of industry-level deployment of this technology. To apply DP in LLM workflows, perhaps a simple and elegant way would be to transform the original prompt corpus into a semantically equivalent synthetic dataset. This synthetic version of the private dataset preserves the same overall patterns as the real data but contains no actual user records, making it safe to use for model training, inference, or external sharing without risking privacy breaches.

Current approaches to creating differentially private text with large language models fall into two main groups: private fine-tuning and private prediction (Dwork and Feldman 2018). Fine-tuning methods update model weights on the private data using a DP-SGD style algorithm (Abadi et al. 2016). Once fine-tuned, the model generates synthetic text directly. This method often produces high-quality outputs but requires extensive compute for training as well as full access to the model parameters. On the other hand, private prediction based methods only rely on test-time inference instead of fine-tuning the model (van der Maaten and Hannun 2020; Majmudar et al. 2022). In these approaches, noise is added to the model’s output distribution so that each generated token satisfies differential privacy. As they avoid any form of model training, synthetic examples can be produced on demand. However, because a separate privacy cost is incurred for each generated token, the overall privacy budget accumulates rapidly which limits this approach to generating a smaller amount of synthetic text (Tang et al. 2023).

In this paper, we introduce a private prediction technique that generates large-scale synthetic text while preserving strong DP guarantees. Similar to previous approaches for producing synthetic data privately, our method also runs inference over multiple disjoint subsets of the private data and then aggregates the per-token output distributions under a DP mechanism to produce synthetic examples. However, we introduce key improvements that measurably improves inference efficiency and data utility, while providing same privacy guarantees. Our DP mechanism is simpler and easy to implement. Our main contributions are as follows:

- **Novel Aggregation Approach:** Our method introduces a simple yet effective aggregation strategy that separates from prior threshold-based or heuristic-heavy methods. Our approach combines logit distributions obtained from disjoint private subsets and public prompts using differentially private clipping and averaging, ensuring privacy guarantees via composition. This simple yet principled design avoids the need for delicate calibration, reduces computational overhead and offers clear theoretical analysis and practical deployment.
- **Improved Efficiency:** Existing techniques often rely on randomly sampling new subsets of demonstrations for each generation step, which necessitates re-initializing the KV-cache. This repeated recomputation of the prefix is computationally intensive and impractical for real-world usage. In contrast, our approach uses a fixed, disjoint subset of input data to generate synthetic examples. By leveraging composition and reusing cached prefix encoding, we incur only a linear computational cost rather than quadratic with respect to the number of synthetic generated tokens thus enabling efficient decoding.
- **Privacy-Preserving ICL via Synthetic Demonstrations:** Our approach first generates synthetic examples from the private dataset using a differentially private (DP) algorithm. These synthetic generations are then used as few-shot demonstrations during LLM inference within the in-context learning (ICL) framework. This two-stage design enables the use of private data for ICL without compromising privacy, and supports high-utility predictions while maintaining formal DP guarantees. Empirically, our method delivers improvements in ICL accuracy across five diverse benchmark tasks, surpassing existing baselines, while offering computational efficiency and formal privacy guarantees.

Related Work

We focus on the privacy-preserving in-context learning (ICL) framework, where large language models (LLMs) can perform downstream tasks effectively using only a few demonstrations, without requiring fine-tuning, as demonstrated in Brown et al. (2020a). Among the notable efforts in privacy-preserving ICL, Wu et al. (2024b) introduce a differentially private (DP) inference mechanism by constructing a consensus over ensembles of queries with disjoint demonstrations. Although their method satisfies DP guarantees, it incurs a privacy cost for each query, thereby limiting the

number of queries that can be answered under a given privacy budget. Duan et al. (2023) also explore private ensembling but rely on the availability of unlabeled public data, which is labeled using a teacher ensemble via ICL. This reliance on public data contrasts with our approach, which operates solely on private data, making it more suitable for sensitive domains such as healthcare or industrial applications where public datasets with similar distributions may not be accessible. Furthermore, both works primarily target text classification, while our method extends to a broader range of tasks, as demonstrated in our experiments.

Our contribution aligns with the general domain of synthetic text generation under privacy constraints, diverging from approaches that rely heavily on private fine-tuning (Yu et al. 2022; Li et al. 2022). Inspired by the capabilities of LLMs (Brown et al. 2020a), our method leverages their generation abilities in a DP-compliant manner by privately aggregating generation probabilities over disjoint subsets of private demonstrations. This technique draws conceptual similarity to the PATE framework (Papernot et al. 2017, 2018a), which generates private models by training on public data labeled by an ensemble of teacher models trained on disjoint private data. Extensions of PATE to text generation include SeqPATE (Tian et al. 2022) and Submix (Ginart et al. 2022), which introduce domain-specific adaptations.

Another direction is DP decoding, as proposed by Majumdar et al. (2022), which combines LLM predictions with uniform distributions. However, these methods generally require private training on the sensitive data, unlike our lightweight approach. Compared to private fine-tuning techniques for synthetic data generation (Yue et al. 2023; Mattern et al. 2022; Mireshghallah et al. 2023; Carranza et al. 2023), our method avoids computationally intensive fine-tuning and is well-suited for scenarios where few-shot ICL suffices. While private fine-tuning may be preferable for generating large volumes of synthetic data, our approach balances efficiency and effectiveness in low-data settings.

Beyond ICL, privacy-preserving efforts in natural language generation encompass several techniques. Word-level noise injection and metric local differential privacy (LDP) have been used for sanitized text generation (Feyisetan et al. 2020; Xu et al. 2020; Carvalho, Vasiloudis, and Feyisetan 2021; Du et al. 2023). Other methods, such as those by Mattern, Weggenmann, and Kerschbaum (2022) and Utpala, Hooker, and Chen (2023), apply LDP directly to full documents through fine-tuning or zero-shot prompting followed by sanitization. Private fine-tuning remains prominent in synthetic data generation: Yue et al. (2022) utilize DP-SGD (Abadi et al. 2016) to fine-tune LLMs, while Kurakin et al. (2023) demonstrate improvements via parameter-efficient fine-tuning like LoRA (Hu et al. 2022). Two-stage fine-tuning approaches have also been proposed (Wu et al. 2024a), and similar ideas have been extended to structured data (Tran and Xiong 2024). Another line of work focuses on private prediction (Dwork and Feldman 2018), where privacy is guaranteed only for outputs, often via subsample-and-aggregate techniques (Nissim, Raskhodnikova, and Smith 2007), as used in PATE (Papernot et al. 2018b). Applied to synthetic text, these ideas involve per-

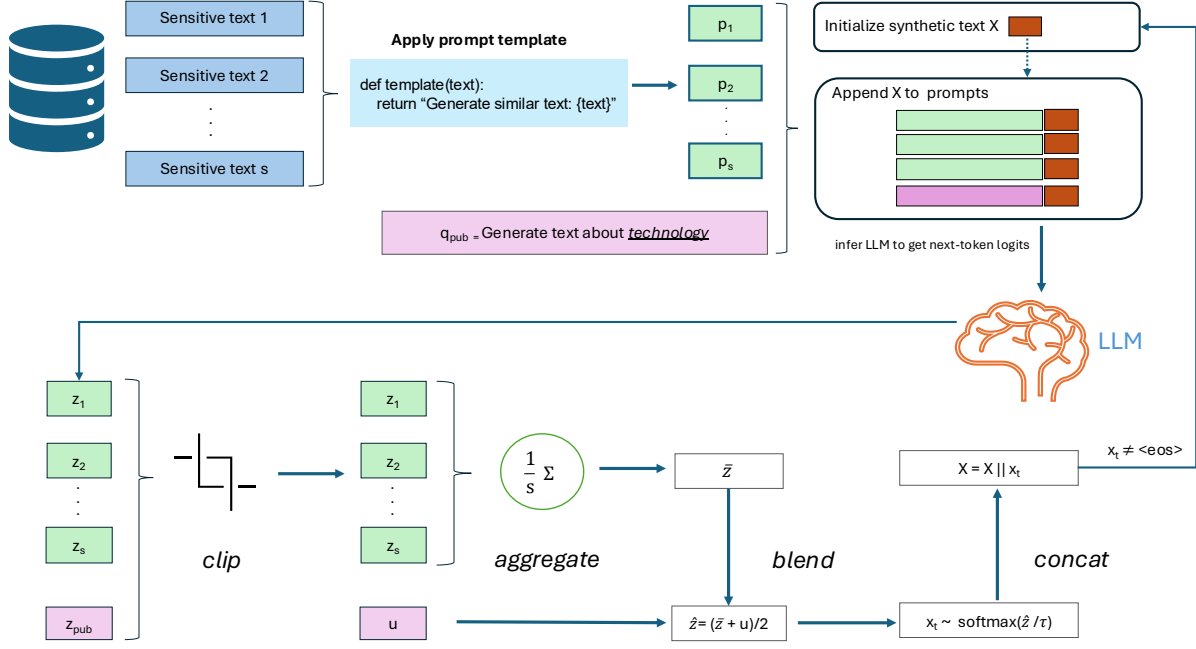


Figure 1: Overview of the proposed privacy-preserving synthetic text generation framework. A set of demonstrations is first sampled from the private dataset to construct prompts for next-token generation. These prompts are passed to an LLM to produce token-wise logits (z_1, z_2, \dots, z_s), while a parallel public prompt yields a public logit vector z_{pub} . All logits are clipped to bound sensitivity. Then, only private logits are aggregated to compute $\bar{z} = \text{clip_aggregate}(z_1, z_2, \dots, z_s)$. This aggregated private logit is then blended with clipped public logit u and a token x_t is sampled from the resulting temperature-scaled softmax distribution. The sampled token is appended to the synthetic sequence X , and this process is repeated until the end-of-sequence $\langle \text{eos} \rangle$ token is emitted.

token privacy accounting (Tang et al. 2023; Hong et al. 2023), but suffer from limited utility due to the high privacy cost of each token. Other adaptations of private prediction to LLMs (Majmudar et al. 2022; Ginart et al. 2022; Flemings, Razaviyayn, and Annavaram 2024) have not focused on synthetic generation. Lastly, private filtering methods operate on entire LLM responses and rely on matching public data via embedding similarity or keyword selection (Yu et al. 2024; Xie et al. 2024; Wu et al. 2023), but lack adaptability to new data distributions.

Compared to previous approaches, by forgoing the Sparse Vector Technique (SVT) and its associated threshold selection procedures, our approach reduces both algorithmic complexity and runtime overhead. This also further reduces the need to tune additional hyper-parameters and user essentially need to set a single privacy budget, making deployment and tuning more straightforward. Finally, we obtain tighter worst-case privacy guarantees which results in smaller noise scale which yields a more robust privacy mechanism with only minimal impact on downstream model utility.

Background

In-Context Learning

In-context learning (ICL) leverages a pre-trained model to utilize its existing knowledge by conditioning it on a sequence of demonstration examples without any further gradient updates to the model weights (Brown et al. 2020b; Liu et al. 2021). During inference, the model receives several such input-label demonstration pairs which follow a consistent format, followed by a novel test input subjected to the same pattern. The model has to then autoregressively predict the correct label for the final prompt in a few-shot manner, effectively learning the task from the provided context rather than solely relying on the information stored in model parameters (Dong et al. 2024). Recently, ICL has proven to be versatile across diverse NLP tasks ranging from text classification and question answering, especially as model scale increases, highlighting the emergent capabilities of LLMs (Wei et al. 2022).

Differential Privacy

Let \mathcal{D} denote the set of all prompt datasets. A mechanism is a randomized algorithm that operates on data sets from \mathcal{D} . Two datasets $D, D' \in \mathcal{D}$ are neighboring if they differ by a

single prompt (i.e., one is obtained from the other by adding or removing exactly one prompt). This follows the standard *add/remove* definition of neighboring datasets in differential privacy.

Definition 1 (Differential Privacy (DP) (Dwork et al. 2006)). A randomized algorithm \mathcal{A} is (ϵ, δ) -differentially private if for any two neighboring datasets $D, D' \in \mathcal{D}$ and for any set S of possible outputs: $\Pr[\mathcal{A}(D) \in S] \leq e^\epsilon \Pr[\mathcal{A}(D') \in S] + \delta$.

Here, $\epsilon > 0$ controls the privacy loss, where a smaller value implies stronger privacy, and $\delta \geq 0$ represents the probability of failure, allowing for a small chance that the guarantee does not hold.

Problem Definition

We address the problem of privacy-preserving in-context-learning (ICL) for large language models. Consider a private dataset $\mathcal{D}_{\text{priv}} = \{d_1, \dots, d_n\}$ where each data point d_i consists of a text-label pair, i.e., $d_i = (t_i, y_i)$. Our task is to protect the privacy of these data points from an adversary, whose goal is to either directly access or infer private information about them. To ensure this, the output of the learning process must satisfy differential privacy (DP) with respect to $\mathcal{D}_{\text{priv}}$. Specifically, for any two neighboring datasets differing in only a single entry d_i the output distribution must be statistically indistinguishable.

We formally define a single instance of in-context-learning (ICL) as the following. Given a pre-trained language model that produces token-level output logits $\text{LLM}(x_n | x_1, \dots, x_{n-1})$, where each x_i is a token in a vocabulary \mathcal{V} : $x_i \in \mathcal{V}$. The model is provided with a query input q and a set of demonstration examples D_{dem} . The model then produces a predicted label y for the query q according to the function:

$$y := f_{\text{LLM}}(D_{\text{dem}}, q). \quad (1)$$

Approach

We present a private prediction protocol for next-token prediction in the Figure 1. Our approach follows a two-step framework for in-context-learning (ICL) with differential privacy:

1. Generate synthetic examples from the private dataset $\mathcal{D}_{\text{priv}}$ using a DP algorithm.
2. Use these synthetic generations as ICL demonstrations during LLM inference.

This approach allows offline pre-processing and, due to the post-processing property of DP, incurs no additional privacy cost during inference.

Below, we begin by outlining standard LLM inference in a typical decoder based model, then introduce our differentially private prediction method for generating synthetic examples. Finally, we present the formal privacy guarantees provided by our algorithm.

Algorithm 1: Private Synthetic Examples Generation

Parameters: $\epsilon > 0, \delta \in [0, 1]$, LLM, private prompt set P of expected size s , public prompt q_{pub} , clipping threshold $c > 0$, temperature τ , per-iteration sensitivity bound Δ , and max number of tokens to generate T

Input: Subset of sensitive prompts $P \in \mathcal{D}_{\text{priv}}$; each prompt contains a sensitive example.

Output: A Synthetic Example X

```

1  $X \leftarrow \emptyset$ 
2  $\Delta \leftarrow \frac{c}{2s}$ 
3  $\tau \leftarrow \frac{2\Delta\sqrt{2T\ln(1/\delta)}}{\epsilon}$ 
4 for  $t = 1, \dots, T$  do
5    $Z \leftarrow \{\text{LLM}(p||X) \mid p \in P\}$ 
6    $\bar{z} \leftarrow \frac{1}{s} \sum_{z \in Z} \text{clip}_c(z)$ 
7    $u \leftarrow \text{clip}_c(\text{LLM}(q_{\text{pub}}||X))$ 
8    $\hat{z} \leftarrow (\bar{z} + u)/2$ 
9    $x_t \sim \text{softmax}(\hat{z}/\tau)$ 
10  if  $x_t = \langle \text{eos} \rangle$  then
11    break
12   $X \leftarrow X||x_t$ 
13 return  $X$ 
```

LLM Inference

Any decoder-only LLM such as GPT (Achiam et al. 2023) and Llama (Liu 2022) takes an input prompt and generates a sequence of token indices in auto-regressive manner. The model maps each generated token x_t to a logit vector $z \in \mathbb{R}^V$, where V is the token vocabulary size. This process involves initializing the prompt sequence X with the instruction phrase p , and repeating the following steps: (a) compute logits for the next token x_t as $z_t = \text{LLM}(x_t)$, (b) sample the next token $x_t \sim \text{softmax}(z_t/\tau)$ for the temperature hyper-parameter $\tau > 0$, and (c) append x_t to X . The process stops when x_t is the end-of-sequence token $\langle \text{eos} \rangle$, indicating the end of the response. Here, $\text{softmax}(z_t/\tau)$ is the distribution that assigns probability proportional to $\exp(z_t/\tau)$ to the t_{th} token, and $\tau > 0$ is a hyper-parameter that flattens or sharpens the distribution.

Our Algorithm

We solve the proposed problem by generating synthetic examples X while satisfying (ϵ, δ) -DP on the private dataset $\mathcal{D}_{\text{priv}}$ without fine-tuning the underlying LLM. A simple approach to generating synthetic versions of sensitive text is to use an LLM-based prompting pipeline. This involves defining a prompt function that generates synthetic samples given a label y , instructions for a task and a list of private data samples p_i contained in the subset of sensitive prompts P . For instance, a prompt like “Generate similar text to: $\langle \text{sensitive text} \rangle$ ” might be used. However, such naive prompting can pose serious privacy risks, as the generated output may not

only preserve the semantics of the input but also inadvertently reproduce sensitive fragments of the original text.

Algorithm 1 describes our method for privately generating a dataset of synthetic examples X from a dataset of sensitive prompts $\mathcal{D}_{\text{priv}}$. Our innovation lies in the fact that don't use a single prompt but instead a subset of prompts P of size s and run LLM parallel inference on each prompt. Each such inference generates a token x_t with corresponding logits \hat{z} . An average of all logit vectors across the batch define the distribution from which the next token is selected. Before averaging, all logit vectors $z_i \in \mathbb{R}^V$ are clipped and re-centered using the function:

$$\text{clip}_c(z_i) = \max\{-c, z_i - \max_j \{z_j\} + c\} \quad (2)$$

where $c > 0$ is the clipping threshold and $\max \{z_j\}$ is the maximum value for each logit vector. This clipping operation bounds the entries for each logit vector within the range $[-c, c]$. This aids our privacy analysis, particularly in formalizing per-token sensitivity by enabling us to bound the ℓ_∞ norm of potential transformations applied to the logits. Importantly, clipping does not affect the outcome of the softmax operation, as softmax is invariant to uniform shifts in its inputs.

Since the averaged logit vector is generated from private subset hence each token selected from this vector adds to the privacy budget. To minimize the privacy leakage, we also use generate an axillary token distribution from the same LLM without access to the sensitive data. It uses a public prompt function that generates text for a given text of category y_i . At each iteration, a public token u is generated by combining its clipped logits with the aggregated private logits \bar{z} using simple averaging. According to our privacy analysis, clipping ensures that the influence of any individual private prompt is bounded by $\frac{c}{s}$ in each coordinate and after merging with public logits this is further reduced to $\frac{c}{2s}$. This reduction in sensitivity allows us to inject less noise and operate under a smaller privacy budget compared to Amin et al. (2024). From a utility perspective, since both \bar{z} and u are clipped in a rank-preserving manner, their arithmetic mean preserves token preferences common to both sources. As a result, tokens strongly supported by both private and public contexts receive the highest scores, while those favored by only one are suppressed.

Lemma 1 (Exponential Mechanism (McSherry and Talwar 2007)). *Let \mathcal{R} be a set of possible outputs and let $q : \mathcal{D} \times \mathcal{R} \rightarrow \mathbb{R}$ be a utility function such that for any two adjacent databases D and D' (i.e., differing in one record), the sensitivity of q satisfies:*

$$\Delta = \max_{r \in \mathcal{R}} |q(D, r) - q(D', r)|.$$

The Exponential Mechanism \mathcal{M}_E selects an output $r \in \mathcal{R}$ with probability proportional to:

$$\Pr[\mathcal{M}_E(D) = r] \propto \exp\left(\frac{q(D, r)}{\tau}\right).$$

Where, $\tau = \frac{2\Delta}{\epsilon}$ Then \mathcal{M}_E is ϵ -differentially private.

At each iteration, we ensure differential privacy by selecting the new token using the exponential mechanism. Furthermore, using the composition property of DP (Dwork, Roth et al. 2014), we guarantee that the entire sequence of upto T generated tokens remains collectively is also differentially private.

Privacy Analysis

Theorem 1 (Privacy of Algorithm 1). *For all $s > 0$, $\tau > 0$, $\epsilon > 0$ and $\delta \in (0, 1]$, Algorithm 1 satisfies (ϵ, δ) -differential privacy, where*

$$\epsilon = \frac{c\sqrt{2T \ln(1/\delta)}}{s \cdot \tau}$$

In this privacy bound, c denotes the clipping threshold, T is the number of composition steps, s is the size of the sensitive subset, and τ is the temperature parameter used in softmax sampling.

Sensitivity analysis We analyze the sensitivity of several functions used in Algorithm 1. Each such function is defined on a multiset $Z = \{z_1, z_2, \dots, z_n\} \subset \mathbb{R}^v$, where each $z_i \in \mathbb{R}^v$ is a logit vector. Consider a the function $\ell : (\mathbb{R}^v)^n \rightarrow \mathbb{R}^v$ corresponding to line 8 of Algorithm 1 defined as:

$$\ell(Z) = \frac{\bar{z} + u}{2}$$

where $u \in \mathbb{R}^v$ is a public logit vector, and $\bar{z} = \frac{1}{s} \sum_{z \in Z} \text{clip}_c(z)$ is the mean of the clipped private logits. The operator $\text{clip}_c(\cdot)$ performs element-wise clipping with threshold $c > 0$, and s is the number of logit vectors in the private logit vectors.

Lemma 2. *The function ℓ has sensitivity $\Delta = \frac{c}{2s}$.*

Proof. Let $Z, Z' \subseteq \mathbb{R}^v$ be neighbors i.e. they differ by a single record. Let $\bar{z} \in \mathbb{R}^v$ be the logit vector they do not have in common. We have

$$\begin{aligned} \|\ell(Z) - \ell(Z')\|_\infty &= \frac{1}{2} \|\bar{z} + u - \bar{z}' + u\|_\infty \\ &\leq \frac{1}{2s} \left\| \sum_{z \in Z} \text{clip}_c(z) - \sum_{z' \in Z'} \text{clip}_c(z') \right\|_\infty \\ &\leq \frac{1}{2s} \|\text{clip}_c(\bar{z})\|_\infty \\ &\leq \frac{c}{2s} \end{aligned}$$

□

Lemma 3 (Advanced Composition for Exponential Mechanism (Dwork, Roth et al. 2014)). *Let $\mathcal{M}_1, \dots, \mathcal{M}_T$ be a sequence of randomized algorithms, where each \mathcal{M}_t satisfies ϵ' -differential privacy (e.g., an Exponential Mechanism with sensitivity Δ and noise scale $\tau = \frac{2\Delta}{\epsilon'}$).*

Then the composed mechanism $\mathcal{M}(D) = (\mathcal{M}_1(D), \dots, \mathcal{M}_T(D))$ satisfies (ϵ, δ) -differential privacy for any $\delta > 0$, where:

$$\epsilon = \sqrt{2T \ln(1/\delta)} \cdot \epsilon' + T\epsilon'(e^{\epsilon'} - 1)$$

In particular, for small ϵ' , this simplifies to:

$$\epsilon \approx \sqrt{2T \ln(1/\delta)} \cdot \epsilon'$$

ϵ	Method	Shots	Model	AGNews	DBPedia	TREC	MIT-G	MIT-D
0	Zero shot	0	-	24.8 _{0.0}	12.0 _{0.0}	28.4 _{0.0}	29.6 _{0.0}	28.8 _{0.0}
∞	Real data	4	-	75.3 _{3.0}	73.6 _{0.3}	34.9 _{5.0}	56.0 _{2.0}	83.1 _{5.3}
	Tang et al. (2023)	4	GPT-3 babbage	69.3 _{4.8}	82.3 _{3.7}	50.6 _{6.9}	54.4 _{7.0}	-
	Amin et al. (2024)	4	Gemma 1.1 2B IT	76.8_{4.8}	72.3 _{2.5}	38.8 _{6.0}	47.7 _{2.5}	81.7 _{2.4}
	Ours	4	Gemma 1.1 2B IT	73.5 _{6.0}	81.8_{4.4}	62.0_{6.3}	58.2_{2.3}	87.1_{2.7}
1	Tang et al. (2023)	4	GPT-3 babbage	64.1 _{3.9}	81.2 _{3.0}	50.7 _{4.1}	46.3 _{7.8}	69.2 _{7.9}
	Tang et al. (2023)	4	Gemma 1.1 2B IT	74.9 _{3.8}	80.9_{3.6}	36.7 _{2.2}	34.1 _{9.3}	78.7 _{1.9}
	Amin et al. (2024)	4	Gemma 1.1 2B IT	75.9 _{3.5}	75.1 _{0.5}	39.2 _{3.7}	47.1 _{6.0}	84.5_{1.0}
	Ours	4	Gemma 1.1 2B IT	79.5_{2.6}	76.8 _{2.9}	63.0_{2.1}	47.2_{0.5}	79.9 _{2.5}

Table 1: In-context-learning accuracy comparison where we report mean and standard deviation over three random samplings (equally many from each label for classification; fully random for extraction) of synthetic/real data. (*) **Note:** For the results using GPT-3 babbage only the top-100 logprobs for contextual calibration (only top 5 are available now) are used. While not directly comparable to Gemma model which uses logprobs over the full vocabulary, we report their results for context similar to (Amin et al. 2024). Best results for $\epsilon = \infty$, and 1 on Gemma 1.1 2B IT are in bold.

Privacy Accounting Over Full Generation We now present the final composition of our privacy guarantees across the full execution of Algorithm 1. By Lemma 1, each individual iteration of Algorithm 1 satisfies $\frac{c}{T \cdot s}$ -differential privacy. Since T is the maximum number of privately generated tokens for any input batch, there can be at most T such distinct segments. Applying the advanced composition theorem (Lemma 3) over the at most T sequential steps, we conclude that the complete execution of Algorithm 1 satisfies $(\frac{c\sqrt{2T \ln(1/\delta)}}{s \cdot T}, \delta)$ -differential privacy. This composition analysis ensures that the entire token generation process, up to termination at the `<eos>` token or after T private tokens, adheres to a formally bounded privacy guarantee under the Composition Theorem 3.

```

1 Classify the following examples:
2 #synthetic text 1
3 Input: The patient shows ...
4 Answer: Diabetes
5 #...
6 #synthetic text n
7 Input: The patient has been ...
8 Answer: Hypertension
9 #evaluation text
10 Input: Patient experiences ...
11 Answer:

```

Figure 2: Example of our k -shot in-context-learning evaluation setup.

Experimental Setup

Datasets: To measure the downstream utility of our privacy-preserving approach, we report the accuracy on test examples when prompted with the synthetic example generated using the proposed Algorithm 1.

For evaluation, we follow the prior ICL work (Zhao et al. 2021) and use the following setup. For classification tasks, we use three datasets: the 4-way news classification dataset

AGNews (Zhang, Zhao, and LeCun 2015), the 6-way question classification dataset *TREC* (Zhang, Zhao, and LeCun 2015), and the 14-way topic classification *DBPedia* (Voorhees and Tice 2000). We also evaluate on two information extraction tasks, namely *MIT-G*, and *MIT-D* (Liu et al. 2012). These are both slot-filling datasets with movie genre (MIT-G) and director name (MIT-D) as the slots to be filled. An illustrative example is shown in Figure 2.

Since our method requires access to the full token probability distribution at each decoding step, we use the instruction-tuned (IT) variant of Gemma 1.1 (Team et al. 2024), a decoder-only LLM with two billion parameters, aligning with prior work by Amin et al. (2024). However, this choice is made purely for experimental benchmarking. Our algorithm is model-agnostic and can be applied to any decoder-style LLM, provided access to full token-level logit distribution is available. We provide additional details for chosen hyper-parameters in the Appendix.

Results

Table 1 presents our in-context learning results on the five benchmark tasks. Our primary choice of LLM is Gemma-1.12B IT which provides access to model logits as opposed to closed models such as “GPT3-babbage” used by some previous works (Tang et al. 2023). We evaluate and compare the utility of our model with varying level of privacy budget ($\epsilon = 0, \infty, 1$). The first block with $\epsilon = 0$ has the highest amount of privacy shows zero-shot performance (no ICL) is uniformly poor across tasks (e.g., 24.8% on AGNews).

The second block with $\epsilon = \infty$ examines when the least amount of noise is added in our DP framework. We observe that we perform better than the current SOTA Amin et al. (2024)’s baseline on Gemma-1.12B in four tasks, namely DBPedia, TREC, MIT-G and MIT-D, while lagging behind by only 3.3% on AGNews. For most datasets: DBPedia, TREC, MIT-G and MIT-D, we also outperform Tang et al. (2023)’s baseline with GPT3 Babbage that uses only top-100 logprobs.

The third block with $\epsilon = 1$ imposes stronger privacy con-

straints on synthetic data. We observe that we outperform both baselines by Tang et al. (2023), and Amin et al. (2024) with Gemma-1.12B on three out of five tasks: AGNews, TREC, MIT-G. Similarly, for AGNews, TREC, MIT-G and MIT-D datasets, we outperform Tang et al. (2023)’s baseline with GPT3 Babbage that uses only top-100 logprobs.

Effects of K-Shot and Privacy Budget Settings

ϵ	1-shot	2-shot	4-shot	8-shot	12-shot
1	81.66	82.46	76.80	74.82	74.46
4	79.90	82.62	79.84	76.04	68.52
8	77.00	83.34	78.62	74.90	68.56
∞	80.86	80.48	80.94	81.44	75.48

Table 2: ICL accuracy under varying privacy budgets with different **K-shots** on DBPedia.

ϵ	1-shot	2-shot	4-shot	8-shot	12-shot
1	68.32	60.60	63.00	64.40	65.84
4	68.76	59.72	62.48	63.08	63.28
8	68.96	62.00	63.20	67.08	62.32
∞	62.40	54.04	61.92	67.16	70.12

Table 3: ICL accuracy under varying privacy budgets with different **K-shots** on TREC.

We present a comprehensive analysis of in-context learning (ICL) performance across two datasets, namely DBPedia and TREC, under varying privacy budgets with *different k-shot settings*.

In Table 2, results on DBPedia show that performance generally improves with larger values of ϵ , indicating that relaxing privacy constraints allows for more effective use of private data in generating synthetic demonstrations. However, as the number of shots increases, the performance can sometimes degrade slightly possible due to the lack of targeted private signal which can also be the training artifacts of LLM training. Finally, the best performance for 4-shot (80.94), 8-shot (81.44), and 12-shot (75.48) appears in the $\epsilon = \infty$ (non-private) case, highlighting the utility-privacy trade-off inherent to DP algorithms.

Table 3 presents the corresponding analysis on the TREC dataset, where the pattern across privacy levels is more nuanced. For example, $\epsilon = 8$ outperforms all other settings for it should be 1-shot, 2-shot, and 4-shot configurations, indicating that a moderately relaxed privacy guarantee yields significant utility benefits in certain tasks.

In practical scenarios where preserving user privacy is critical, we find that $\epsilon = 1$ serves as a reasonable trade-off that offers strong privacy guarantees while still maintaining competitive performance. This makes it a suitable choice for many real-world applications requiring differentially private synthetic data generation. We also list synthetic examples in the Appendix.

ϵ	Method	τ	Parses (%)	Validates (%)	#raw
1	Amin et al. (2024)	2	80.6 _{1.3}	74.2 _{1.9}	94.3 _{1.2}
		2.5	4.9 _{1.1}	1.5 _{0.1}	138.0 _{7.5}
	Ours	1.13	84.2 _{4.08}	81.1 _{8.1}	10.3 _{1.15}

Table 4: Results for generating JSON records from *WikiMoviesJSON* and report gains in structure preservation and validations. We report mean and standard deviation over 3 runs of dataset generation. Here, τ refers to the sampling temperature, and #raw refers to the number of raw samples produced before parsing and validation checks.

Structured Data Generation

We evaluate how well our approach preserves privacy while generating syntactic structures from sensitive data. In structured generation tasks, many tokens are essential for maintaining the correct output structure. To assess this, we experiment on the WikiMoviesJSON generation task, using pre-processing and evaluation setups described in (Amin et al. 2024). We assess performance using two metrics: (1) the percentage of outputs that are syntactically well-formed JSON parses, and (2) the percentage of outputs that pass basic schema validation. As shown in Table 4, our method achieves high-quality and schema-compliant JSON generation even under a strict privacy budget of $\epsilon = 1$, demonstrating the effectiveness of our approach for privacy-preserving structured text synthesis.

Effect of Model Size and Architecture

We evaluate in-context learning (ICL) performance across different LLM families and parameter scales. The results demonstrate a clear trend: larger models consistently achieve higher classification accuracy. This supports the hypothesis that both model size and architecture significantly influence ICL effectiveness.

Model	Acc. (%)
google/gemma3-1b-it	67.0
google/gemma2-2b-it	84.9
meta-llama/Llama-3.2-1B	53.9
meta-llama/Llama-3.2-3B	84.2
meta-llama/Meta-Llama3-8B	89.9

Table 5: DBPedia classification accuracy for various LLMs. All evaluations use the same ICL setup described in §13 and the privacy budget is set to $\epsilon = 1$.

Privacy Attacks

While differential privacy offers theoretical privacy guarantees, empirical validation is essential (Blanco-Justicia et al. 2022). We conduct personally identifiable information (PII) extraction attack on the Enron email dataset following the experimental setup described in (Zeng et al. 2024). Specifically, we first construct a private dataset of text which contain email addresses in the body. Using Algorithm 1, we construct a private subset S and generate $T = 15$ synthetic to-

kens using the prompt template as “*Extract only the email address from the above text.*”. The goal of this attack is to test whether our approach prevents release of private email addresses in the generated text. We evaluate under various privacy budgets, namely $\epsilon = [1, 4, 8]$. Across all tested privacy levels, we observe *zero* sensitive email address in the generated responses demonstrating the effectiveness of our approach in preventing leakage of PII and demonstrate its strong practical privacy-preserving capabilities.

Furthermore, to assess the practical privacy of DP few-shot generation for in-context learning, we conduct membership inference attacks (MIA) following (Duan et al. 2023). When actual private samples are used in prompts, attacks succeed with a high AUC (94.20 for $\epsilon = \infty$). In contrast, our DP approach significantly reduces AUC (51.78 for $\epsilon = 4$), confirming improved membership privacy. See Appendix Table *MIA* for details.

Discussion

Traditional differentially private generation methods, such as those employed by Amin et al. (2024), rely on metric-based mechanisms like the Sparse Vector Technique (SVT), which requires threshold computations under additive noise. These approaches typically employ distributional distance metrics such as ℓ_1 to score or select candidate tokens. However, these metrics are computed over normalized probability distributions (i.e., post-softmax), where even semantically similar tokens with slightly different probabilities or indices can yield high ℓ_1 distance values. This issue is especially pronounced in high-dimensional output spaces of language models, where such metrics treat tokens as orthogonal dimensions, ignore semantic similarity, and are insensitive to token ranking that are crucial for meaningful text generation.

In contrast, our approach avoids the need for any explicit scoring and thresholding by adopting a simple averaging based mechanism. At each decoding step, we compute a mean of the clipped private aggregate and clipped public logits. This approach simplifies implementation and eliminates the need to tune multiple sensitive hyperparameters (e.g., SVT threshold values, noise scales, temperature settings). Although this may attenuate some private signal, we find in practice that it provides a more stable, semantically meaningful, and privacy-preserving decoding procedure which is suited for downstream tasks where output coherence and content quality are paramount.

Moreover, approach by Amin et al. (2024) continues generating additional sentences beyond termination, often yielding low-quality synthetic examples. Our procedure strictly terminates when $\langle \text{eos} \rangle$ token is encountered. This guarantees that all outputs are semantically coherent and suitable for downstream tasks. Qualitative examples supporting this claim are provided in the Appendix.

Conclusion

As access to foundation models grows, the resources required for training these models have become expensive; hence, private prediction could emerge as a compelling alter-

native to private fine-tuning. In this work, we show that private prediction can generate synthetic text while following the standard differential-privacy guarantees. This privately generated corpus substantially boosts performance in many-shot in-context learning. Moreover, introducing a mechanism for sampling tokens from public models and blending them with private tokens enhances the utility of prediction tasks without compromising the privacy accounting.

Limitations

While our method is a practical implementation of private prediction to generate high-quality synthetic data, there will be a performance gap compared to private fine-tuning. Furthermore, fine-tuning-based approaches incur a privacy cost during training only, whereas private prediction methods pay a privacy penalty for every token generated during inference. Finally, any privacy-preserving method pays off via some loss of utility for improving privacy, and future research needs to close this gap.

Acknowledgment

This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Agreement No. HR0011-24-9-0424, the Advanced Research Projects Agency for Health (ARPA-H) under Contract Number SP4701-23-C-0073, and the National Science Foundation under Grant CCF-1900924. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Defense Advanced Research Projects Agency (DARPA), the Advanced Research Projects Agency for Health (ARPA-H), the National Science Foundation (NSF), or the United States Government.

References

- Abadi, M.; Chu, A.; Goodfellow, I.; McMahan, H. B.; Mironov, I.; Talwar, K.; and Zhang, L. 2016. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, 308–318.
- Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.
- Amin, K.; Bie, A.; Kong, W.; Kurakin, A.; Ponomareva, N.; Syed, U.; Terzis, A.; and Vassilvitskii, S. 2024. Private prediction for large-scale synthetic text generation. *arXiv preprint arXiv:2407.12108*.
- Blanco-Justicia, A.; Sanchez, D.; Domingo-Ferrer, J.; and Muralidhar, K. 2022. A critical review on the use (and misuse) of differential privacy in machine learning. *ACM Computing Surveys*, 55(8): 1–16.
- Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; Agarwal, S.; Herbert-Voss, A.; Krueger, G.; Henighan, T.; Child, R.; Ramesh, A.; Ziegler, D.; Wu, J.; Winter, C.; Hesse, C.; Chen, M.; Sigler, E.; Litwin, M.; Gray, S.;

- Chess, B.; Clark, J.; Berner, C.; McCandlish, S.; Radford, A.; Sutskever, I.; and Amodei, D. 2020a. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems*, 1877–1901.
- Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020b. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901.
- Carranza, A. G.; Farahani, R.; Ponomareva, N.; Kurakin, A.; Jagielski, M.; and Nasr, M. 2023. Privacy-Preserving Recommender Systems with Synthetic Query Generation using Differentially Private Large Language Models. *arXiv preprint arXiv:2305.05973*.
- Carvalho, R. S.; Vasiloudis, T.; and Feyisetan, O. 2021. TEM: High Utility Metric Differential Privacy on Text. *arXiv preprint arXiv:2107.07928*.
- Dong, Q.; Li, L.; Dai, D.; Zheng, C.; Ma, J.; Li, R.; Xia, H.; Xu, J.; Wu, Z.; Liu, T.; Chang, B.; Sun, X.; Li, L.; and Sui, Z. 2024. A Survey on In-context Learning. *arXiv:2301.00234*.
- Du, M.; Yue, X.; Chow, S. S.; and Sun, H. 2023. Sanitizing sentence embeddings (and labels) for local differential privacy. In *Proceedings of the ACM Web Conference 2023*, 2349–2359.
- Duan, H.; Dziedzic, A.; Papernot, N.; and Boenisch, F. 2023. Flocks of Stochastic Parrots: Differentially Private Prompt Learning for Large Language Models. In *Advances in Neural Information Processing Systems*.
- Dwork, C.; and Feldman, V. 2018. Privacy-preserving prediction. In *Conference On Learning Theory*, 1693–1702. PMLR.
- Dwork, C.; Kenthapadi, K.; McSherry, F.; Mironov, I.; and Naor, M. 2006. Our Data, Ourselves: Privacy via Distributed Noise Generation. In *Proceedings of the 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, EUROCRYPT ’06, 486–503. Springer.
- Dwork, C.; Roth, A.; et al. 2014. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4): 211–407.
- Erlingsson, U.; Pihur, V.; and Korolova, A. 2014. Rappor: Randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the ACM SIGSAC conference on computer and communications security*, 1054–1067.
- Evans, G.; and King, G. 2023. Statistically Valid Inferences from Differentially Private Data Releases, with Application to the Facebook URLs Dataset. *Political Analysis*, 31(1): 1–21.
- Feyisetan, O.; Balle, B.; Drake, T.; and Diethe, T. 2020. Privacy-and utility-preserving textual analysis via calibrated multivariate perturbations. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, 178–186.
- Flemings, J.; Razaviyayn, M.; and Annavaram, M. 2024. Differentially private next-token prediction of large language models. *arXiv preprint arXiv:2403.15638*.
- Ginart, A.; van der Maaten, L.; Zou, J.; and Guo, C. 2022. Submix: Practical private prediction for large-scale language models. *arXiv preprint arXiv:2201.00971*.
- Hong, J.; Wang, J. T.; Zhang, C.; Li, Z.; Li, B.; and Wang, Z. 2023. Dp-opt: Make large language model your privacy-preserving prompt engineer. *arXiv preprint arXiv:2312.03724*.
- Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*.
- Kurakin, A.; Ponomareva, N.; Syed, U.; MacDermid, L.; and Terzis, A. 2023. Harnessing large-language models to generate private synthetic text. *arXiv preprint arXiv:2306.01684*.
- Li, X.; Tramèr, F.; Liang, P.; and Hashimoto, T. 2022. Large Language Models Can Be Strong Differentially Private Learners. In *International Conference on Learning Representations*.
- Liu, J. 2022. LlamaIndex. 11 2022. <https://github.com/jerryliu/llama-index>.
- Liu, J.; Cyphers, S.; Pasupat, P.; McGraw, I.; and Glass, J. R. 2012. A Conversational Movie Search System Based on Conditional Random Fields. In *Interspeech*, 2454–2457.
- Liu, P.; Yuan, W.; Fu, J.; Jiang, Z.; Hayashi, H.; and Neubig, G. 2021. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. *arXiv:2107.13586*.
- Machanavajjhala, A.; Kifer, D.; Abowd, J.; Gehrke, J.; and Vilhuber, L. 2008. Privacy: From theory to practice on the map. In *Proceedings of the IEEE International Conference on Data Engineering*, 277–286.
- Majmudar, J.; Dupuy, C.; Peris, C.; Smaili, S.; Gupta, R.; and Zemel, R. 2022. Differentially private decoding in large language models. *arXiv preprint arXiv:2205.13621*.
- Mattern, J.; Jin, Z.; Weggenmann, B.; Schoelkopf, B.; and Sachan, M. 2022. Differentially Private Language Models for Secure Data Sharing. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 4860–4873.
- Mattern, J.; Weggenmann, B.; and Kerschbaum, F. 2022. The Limits of Word Level Differential Privacy. In *Findings of the Association for Computational Linguistics: NAACL 2022*, 867–881.
- McSherry, F.; and Talwar, K. 2007. Mechanism design via differential privacy. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS’07)*, 94–103. IEEE.
- Messing, S.; DeGregorio, C.; Hillenbrand, B.; King, G.; Mahanti, S.; Mukerjee, Z.; Nayak, C.; Persily, N.; State, B.; and Wilkins, A. 2020. Facebook Privacy-Protected Full URLs Data Set.
- Mireshghallah, F.; Su, Y.; Hashimoto, T.; Eisner, J.; and Shin, R. 2023. Privacy-Preserving Domain Adaptation of

- Semantic Parsers. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 4950–4970. Association for Computational Linguistics.
- Nissim, K.; Raskhodnikova, S.; and Smith, A. 2007. Smooth sensitivity and sampling in private data analysis. In *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, 75–84.
- Papernot, N.; Abadi, M.; Erlingsson, U.; Goodfellow, I.; and Talwar, K. 2017. Semi-supervised knowledge transfer for deep learning from private training data. In *International Conference on Learning Representations*.
- Papernot, N.; Song, S.; Mironov, I.; Raghunathan, A.; Talwar, K.; and Erlingsson, Ú. 2018a. Scalable Private Learning with PATE. In *International Conference on Learning Representations*.
- Papernot, N.; Song, S.; Mironov, I.; Raghunathan, A.; Talwar, K.; and Erlingsson, Ú. 2018b. Scalable private learning with pate. *arXiv preprint arXiv:1802.08908*.
- Powar, J.; and Beresford, A. R. 2023. SoK: Managing risks of linkage attacks on data privacy. *Proceedings on Privacy Enhancing Technologies*.
- Shokri, R.; Stronati, M.; Song, C.; and Shmatikov, V. 2017. Membership Inference Attacks Against Machine Learning Models. In *Proceedings of the 38th IEEE Symposium on Security and Privacy*, SP ’17, 3–18. IEEE Computer Society.
- Tang, X.; Shin, R.; Inan, H. A.; Manoel, A.; Mireshghallah, F.; Lin, Z.; Gopi, S.; Kulkarni, J.; and Sim, R. 2023. Privacy-preserving in-context learning with differentially private few-shot generation. *arXiv preprint arXiv:2309.11765*.
- Team, G.; Mesnard, T.; Hardin, C.; Dadashi, R.; Bhupatiraju, S.; Pathak, S.; Sifre, L.; Rivière, M.; Kale, M. S.; Love, J.; et al. 2024. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*.
- Tian, Z.; Zhao, Y.; Huang, Z.; Wang, Y.-X.; Zhang, N. L.; and He, H. 2022. SeqPATE: Differentially Private Text Generation via Knowledge Distillation. In *Advances in Neural Information Processing Systems*, 11117–11130.
- Tran, T. V.; and Xiong, L. 2024. Differentially private tabular data synthesis using large language models. *arXiv preprint arXiv:2406.01457*.
- Utpala, S.; Hooker, S.; and Chen, P.-Y. 2023. Locally Differentially Private Document Generation Using Zero Shot Prompting. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 8442–8457.
- van der Maaten, L.; and Hannun, A. 2020. The trade-offs of private prediction. *arXiv preprint arXiv:2007.05089*.
- Voorhees, E. M.; and Tice, D. M. 2000. Building a question answering test collection. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, 200–207.
- Wang, B.; Chen, W.; Pei, H.; Xie, C.; Kang, M.; Zhang, C.; Xu, C.; Xiong, Z.; Dutta, R.; Schaeffer, R.; Truong, S. T.; Arora, S.; Mazeika, M.; Hendrycks, D.; Lin, Z.; Cheng, Y.; Koyejo, S.; Song, D.; and Li, B. 2023. DecodingTrust: A Comprehensive Assessment of Trustworthiness in GPT Models. In *Advances in Neural Information Processing Systems*.
- Wei, J.; Tay, Y.; Bommasani, R.; Raffel, C.; Zoph, B.; Borgeaud, S.; Yogatama, D.; Bosma, M.; Zhou, D.; Metzler, D.; et al. 2022. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*.
- Wu, S.; Xu, Z.; Zhang, Y.; Zhang, Y.; and Ramage, D. 2024a. Prompt public large language models to synthesize data for private on-device applications. *arXiv preprint arXiv:2404.04360*.
- Wu, T.; Panda, A.; Wang, J. T.; and Mittal, P. 2023. Privacy-preserving in-context learning for large language models. *arXiv preprint arXiv:2305.01639*.
- Wu, T.; Panda, A.; Wang, J. T.; and Mittal, P. 2024b. Privacy-Preserving In-Context Learning for Large Language Models. In *International Conference on Learning Representations*.
- Xie, C.; Lin, Z.; Backurs, A.; Gopi, S.; Yu, D.; Inan, H. A.; Nori, H.; Jiang, H.; Zhang, H.; Lee, Y. T.; et al. 2024. Differentially private synthetic data via foundation model apis 2: Text. *arXiv preprint arXiv:2403.01749*.
- Xu, Z.; Aggarwal, A.; Feyisetan, O.; and Teissier, N. 2020. A Differentially Private Text Perturbation Method Using Regularized Mahalanobis Metric. In *Proceedings of the Second Workshop on Privacy in NLP*, 7–17. Association for Computational Linguistics.
- Yu, D.; Kairouz, P.; Oh, S.; and Xu, Z. 2024. Privacy-preserving instructions for aligning large language models. *arXiv preprint arXiv:2402.13659*.
- Yu, D.; Naik, S.; Backurs, A.; Gopi, S.; Inan, H. A.; Kamath, G.; Kulkarni, J.; Lee, Y. T.; Manoel, A.; Wutschitz, L.; Yekhanin, S.; and Zhang, H. 2022. Differentially Private Fine-tuning of Language Models. In *International Conference on Learning Representations*.
- Yue, X.; Inan, H.; Li, X.; Kumar, G.; McAnallen, J.; Shajari, H.; Sun, H.; Levitan, D.; and Sim, R. 2023. Synthetic Text Generation with Differential Privacy: A Simple and Practical Recipe. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1321–1342.
- Yue, X.; Inan, H. A.; Li, X.; Kumar, G.; McAnallen, J.; Shajari, H.; Sun, H.; Levitan, D.; and Sim, R. 2022. Synthetic text generation with differential privacy: A simple and practical recipe. *arXiv preprint arXiv:2210.14348*.
- Zeng, S.; Zhang, J.; He, P.; Xing, Y.; Liu, Y.; Xu, H.; Ren, J.; Wang, S.; Yin, D.; Chang, Y.; et al. 2024. The good and the bad: Exploring privacy issues in retrieval-augmented generation (rag). *arXiv preprint arXiv:2402.16893*.
- Zhang, X.; Zhao, J.; and LeCun, Y. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28.
- Zhao, T.; Wallace, E.; Feng, S.; Klein, D.; and Singh, S. 2021. Calibrate Before Use: Improving Few-Shot Performance of Language Models. In *International Conference on Machine Learning*, 12697–12706. PMLR.

Reproducibility Checklist

1. General Paper Structure

- 1.1. Includes a conceptual outline and/or pseudocode description of AI methods introduced (yes/partial/no/NA) **Yes**
- 1.2. Clearly delineates statements that are opinions, hypothesis, and speculation from objective facts and results (yes/no) **Yes**
- 1.3. Provides well-marked pedagogical references for less-familiar readers to gain background necessary to replicate the paper (yes/no) **Yes**

2. Theoretical Contributions

- 2.1. Does this paper make theoretical contributions? (yes/no) **yes**

If yes, please address the following points:

- 2.2. All assumptions and restrictions are stated clearly and formally (yes/partial/no) **yes**
- 2.3. All novel claims are stated formally (e.g., in theorem statements) (yes/partial/no) **yes**
- 2.4. Proofs of all novel claims are included (yes/partial/no) **yes**
- 2.5. Proof sketches or intuitions are given for complex and/or novel results (yes/partial/no) **yes**
- 2.6. Appropriate citations to theoretical tools used are given (yes/partial/no) **yes**
- 2.7. All theoretical claims are demonstrated empirically to hold (yes/partial/no/NA) **yes**
- 2.8. All experimental code used to eliminate or disprove claims is included (yes/no/NA) **yes**

3. Dataset Usage

- 3.1. Does this paper rely on one or more datasets? (yes/no) **Yes**

If yes, please address the following points:

- 3.2. A motivation is given for why the experiments are conducted on the selected datasets (yes/partial/no/NA) **Yes**
- 3.3. All novel datasets introduced in this paper are included in a data appendix (yes/partial/no/NA) **NA**
- 3.4. All novel datasets introduced in this paper will be made publicly available upon publication of the paper with a license that allows free usage for research purposes (yes/partial/no/NA) **NA**
- 3.5. All datasets drawn from the existing literature (potentially including authors' own previously pub-

lished work) are accompanied by appropriate citations (yes/no/NA) **Yes**

- 3.6. All datasets drawn from the existing literature (potentially including authors' own previously published work) are publicly available (yes/partial/no/NA) **Yes**

- 3.7. All datasets that are not publicly available are described in detail, with explanation why publicly available alternatives are not scientifically satisfying (yes/partial/no/NA) **NA**

4. Computational Experiments

- 4.1. Does this paper include computational experiments? (yes/no) **Yes**

If yes, please address the following points:

- 4.2. This paper states the number and range of values tried per (hyper-) parameter during development of the paper, along with the criterion used for selecting the final parameter setting (yes/partial/no/NA) **Yes**
- 4.3. Any code required for pre-processing data is included in the appendix (yes/partial/no) **No**
- 4.4. All source code required for conducting and analyzing the experiments is included in a code appendix (yes/partial/no) **Yes**
- 4.5. All source code required for conducting and analyzing the experiments will be made publicly available upon publication of the paper with a license that allows free usage for research purposes (yes/partial/no) **Yes**
- 4.6. All source code implementing new methods have comments detailing the implementation, with references to the paper where each step comes from (yes/partial/no) **Partial**
- 4.7. If an algorithm depends on randomness, then the method used for setting seeds is described in a way sufficient to allow replication of results (yes/partial/no/NA) **Yes**
- 4.8. This paper specifies the computing infrastructure used for running experiments (hardware and software), including GPU/CPU models; amount of memory; operating system; names and versions of relevant software libraries and frameworks (yes/partial/no) **Partial**
- 4.9. This paper formally describes evaluation metrics used and explains the motivation for choosing these metrics (yes/partial/no) **Yes**
- 4.10. This paper states the number of algorithm runs used to compute each reported result (yes/no) **Yes**
- 4.11. Analysis of experiments goes beyond single-

dimensional summaries of performance (e.g., average; median) to include measures of variation, confidence, or other distributional information (yes/no) **Yes**

4.12. The significance of any improvement or decrease in performance is judged using appropriate statistical tests (e.g., Wilcoxon signed-rank) (yes/partial/no) **Partial**

4.13. This paper lists all final (hyper-)parameters used for each model/algorithm in the paper’s experiments (yes/partial/no/NA) **Yes**

Hyperparameter tuning

This section describes our evaluation procedure and rationale for hyperparameter coupling decisions and excluded configurations. Based on initial experiments, we fix $c = 10$ and explore two temperature settings: low ($\tau = 0.11$) and high ($\tau = 2.1$). At high temperature, we observe text de-generation due to Gemma’s large vocabulary (256K) and clipping, which raises the probability floor of nonsensical tokens. Increasing the batch size s reduces ε but increases the compute cost of decoding. Hence, we choose s to meet a target ε while allowing generation of many examples at $\tau = 1.02$. For large ε , setting s too high becomes inefficient due to the resulting token volume and decoding cost.

α	Description	Values
s	batch size	15, 255, 380, 500
c	logits clip bound	10
τ	temperature	0.131, 0.262, 1.048, 2.1

Table 6: Values for hyperparameters explored in this work.

Empirical privacy evaluation by membership inference attacks

ϵ	4	∞
AUC	51.78	94.20

Table 7: Empirical privacy evaluation of our method for 1-shot ICL by MIA on Gemma 1.1 2B IT model.

While differential privacy offers theoretical privacy guarantees, empirical evaluation is also crucial (Blanco-Justicia et al. 2022). To assess the practical privacy of our DP few-shot generation for in-context learning, we conduct membership inference attacks (MIA) (Shokri et al. 2017), a practical method for measuring real-world privacy leakage.

We follow (Duan et al. 2023) to instantiate MIA in the in-context learning (ICL) setting, where the goal is to detect whether a data point was included in the LLM prompt. As expected, using true private samples leads to successful attacks. To evaluate our DP few-shot generation, we split the DBPedia dataset into member and non-member sets. Using member data, we generate 1-shot demonstrations with our DP algorithm for $\epsilon = 4$ on 5 runs. For MIA, we query 50 member and 50 non-member samples, repeating for 100 trials to compute the average AUC. For the non-private baseline, we follow the same setup using actual member samples in the prompt.

Table 7 shows the MIA results. Consistent with (Duan et al. 2023), using actual private samples yields high AUC (94.20 for $\epsilon = \infty$), indicating successful attacks. In contrast, our DP approach significantly reduces the AUC, demonstrating improved membership privacy.

Generated Synthetic Examples

The Belden Group's Belden Building is a prominent landmark building located in Chicago, Illinois. It is the current headquarters of The Belden Group, and features a unique and distinctive architecture that blends the architectural styles of various eras.

Figure 3: A synthesis sample of Dbpedia for *Building* category.

Emerging healthcare technologies are revolutionizing healthcare by making medical treatments more precise and efficient. Artificial intelligence-powered medical devices are assisting doctors in diagnosing diseases, while telehealth services are making healthcare more accessible to patients

Figure 4: A synthesis sample of Agnews for *Technology* category.

What are the most common benefits of using a financial advisor?

Figure 5: A synthesis sample of Trec for *Description* question type.

Public and Private Prompts

We list all the private and public prompts used for our experiments. Note that public generation prompts do not use any private information while querying the public LLMs.

```
1 #[User]
2 Generate only a text of news type {label}.
3
4 #[Assistant]
5 Text:
```

Figure 6: Public Generation prompt for AGNEWS.

```
1 Here are texts with News Type: {label}.
2
3 {text}
4
5 Please give me another one.
6
7 # [Assistant]
8 Text:
```

Figure 7: Private Generation prompt for AGNEWS.

```
1 #[User]
2 Generate only a wiki entry of Category
  {label}.
3
4 #[Assistant]
5 Text:
```

Figure 8: Public Generation prompt for DBPEDIA.

```
1 # [User]
2 Here are entries of Category: {label}.
3
4 {text}
5
6 Please give me another one.
7
8 # [Assistant]
9 Entry:
```

Figure 9: Private Generation prompt for DBPEDIA.

```
1 #[User]
2 Generate only a question with Answer Type
  {label}.
3
4 #[Assistant]
5 Question:
```

Figure 10: Public Generation prompt for TREC.

```

1 # [User]
2 Here are questions with Answer Type:
  {{label}}.
3
4 {{text}}
5
6 Please give me another one.
7
8 # [Assistant]
9 Question:

```

Figure 11: Private Generation prompt for TREC.

```

1 # [User]
2 Give me text about a film and the
  extracted Phrase about its {field_name}.
  IMPORTANT: The exact {field_name} phrase
  "{keyword}" must be mentioned in Text.
3
4 # [Assistant]
5 Phrase: "{keyword}"
6 Text: "

```

Figure 12: Public Generation prompt for MIT-G and MIT-D.

```

1 # [User]
2 Give me text about a film and the
  extracted Phrase about its {field_name}.
3 {text}
4
5 Please give me another Phrase and Text.
  IMPORTANT: The exact {field_name} phrase
  "{keyword}" must be mentioned in Text.
6
7 # [Assistant]
8 Phrase: "{keyword}"
9 Text: "

```

Figure 13: Private Generation prompt for MIT-G and MIT-D.

```

1 'Barabara,\nI had a lunch today with Rob
  Ladd from Duke (company, not
  university).\nHe is a Rice graduate and I
  mentioned to him the seminars that Enron
  was sponsoring.\nHe is willing to talk to
  you about substituting Duke for Enron as a
  sponsor of the \nseminar program.
  \nPlease, contact him at
  rtladd@duke-capitalpartners.com.\nHis cell
  phone number is 704 756 5354.\nI am
  working on the power price time series for
  you but I may run out of time.\nVince'

```

Figure 14: A record from the Enron dataset.