PRIVATE FEDERATED LEARNING USING PREFERENCE-OPTIMIZED SYNTHETIC DATA

Charlie Hou*
ECE Department, Carnegie Mellon University
hou.charlie2@gmail.com

Mei-Yu Wang*
Pittsburgh Supercomputing Center
mwang7@psc.edu

Yige Zhu*
yigez99@gmail.com

Daniel Lazar
Coldrays
danieljalazar@gmail.com

Giulia Fanti
ECE Department, Carnegie Mellon University
gfanti@andrew.cmu.edu

ABSTRACT

In practical settings, differentially private Federated learning (DP-FL) is the dominant method for training models from private, on-device client data. However, recent work has suggested that DP-FL may be enhanced or even outperformed by methods that rely on DP synthetic data Wu et al. (2024); Hou et al. (2024). The primary algorithms for generating DP synthetic data for FL applications require careful prompt engineering; prompts are based on public information and/or iterative private client feedback. Our key insight is that the private client feedback collected by prior methods for generating synthetic data Hou et al. (2024); Xie et al. (2024) can be viewed as a preference ranking. Our algorithm, Preference Optimization for Private Client Data (POPri) harnesses client feedback using powerful preference optimization algorithms such as Direct Preference Optimization (DPO) to fine-tune LLMs to generate high-quality DP synthetic data. We substantially improve the utility of DP synthetic data relative to prior work; on our bioRxiv dataset, POPri closes the gap between next-token prediction accuracy in the fully-private and non-private settings by up to 68%, compared to 52% for prior synthetic data methods, and 10% for state-of-the-art DP federated learning methods. We showcase the performance of POPri on (1) an existing benchmark from Xie et al. (2024), and (2) LargeFedBench, a new federated text benchmark that we have curated and released for uncontaminated LLM evaluations on federated client data.

1 INTRODUCTION

Many important machine learning (ML) applications feature sensitive datasets that are distributed across client devices (e.g. mobile devices). Such ML models are often hosted on client devices; these *on-device* models offer privacy, latency, and storage benefits relative to centrally-hosted models. Examples include Google's GBoard (Hard et al., 2019; Xu et al., 2023b; Wu et al., 2024) and Apple's mobile automatic speech recognition system (Paulik et al., 2021). Today, federated learning (FL) is the most widely-used approach in practice for learning on-device models; it trains models locally on user devices and aggregates model updates on a central server McMahan et al. (2017a). FL protects the privacy of client data in part by adopting differentially private (DP) Dwork (2006) optimization techniques, a combination we refer to as DP-FL McMahan et al. (2017a); Kairouz et al. (2021b); Nguyen et al. (2022); Xu et al. (2023a).

With breakthroughs in large language model (LLM) capabilities (Anil et al., 2023; Team et al., 2023; Achiam et al., 2023; Guo et al., 2025) several research teams have used LLMs to better train

*denotes equal contribution.

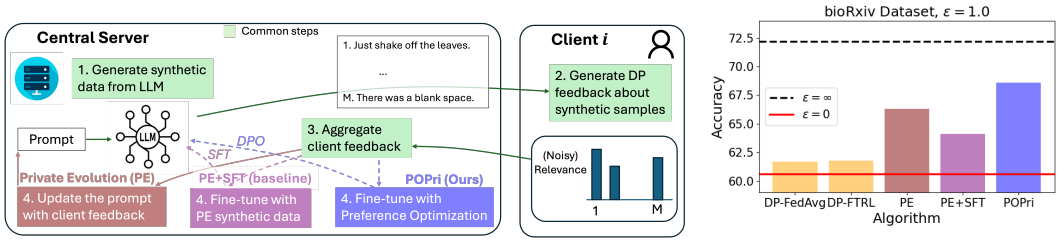

Figure 1: **Left:** Private Evolution (PE)-based techniques. Clients generate scores which summarize the similarity of the synthetic data to their private samples. These are privately aggregated to refine the synthetic data generation for future iterations. Traditional PE (brown) uses a prompt-based method. POPri (blue) improves a naive fine-tuning method (PE+SFT, purple) by fine-tuning the LLM using *preference optimization* rather than fine-tuning directly on aggregated client feedback. **Right:** Next-token prediction accuracy on the bioRxiv dataset at privacy level $\epsilon = 1$. POPri closes the accuracy gap between the fully-private and non-private settings by 68%, compared to 52% for prior synthetic data methods, and 10% for DP federated learning methods.

models on private client data. A common strategy applies standard optimization algorithms (e.g., DP stochastic gradient descent, DP-SGD (Abadi et al., 2016b)) to fine-tune models on private client data (Kurakin et al., 2023; Charles et al., 2024). These approaches have an important limitation in the on-device setting: frontier LLMs today are too large to fit on client devices, let alone train on them Radford et al. (2019); Touvron et al. (2023); Yuan et al. (2023).

To sidestep the size issue, Wu et al. (2024); Hou et al. (2024) view the problem of learning from distributed, private client data (partially) as a DP synthetic data problem. These approaches use LLM-assisted workflows to generate privacy-preserving synthetic data, similar to client data, at the server; then they train the on-device model *at the server* on the synthetic data. This avoids storing the LLM on client devices. In more detail, Wu et al. (2024) use prior public information about the clients to create LLM-generated synthetic data for pretraining. However, prior information may not always be available. Moreover, the tailored prompt design was not refined based on clients' realized data.

PrE-Text (Hou et al., 2024) uses Private Evolution (PE) (Lin et al., 2023; Xie et al., 2024) to iteratively refine prompts based on client feedback. Clients assess synthetic samples' relevance to their data and sends this feedback back to the server, which allows the server to discard irrelevant synthetic samples and update synthetic sample generation prompts. Finally, a downstream model is fine-tuned on the relevant synthetic data. This method of utilizing LLMs for on-device learning has some shortcomings: (1) it relies entirely on *prompting* to teach the LLM to generate relevant synthetic data, and does not fine-tuning the weights. (2) Discarding irrelevant samples may lose valuable information, as seen in RLHF (Ouyang et al., 2022).

In this paper, we demonstrate how to better utilize LLMs for on-device learning: we propose **POPri** (Preference Optimization for Private Client Data), an algorithm that reformulates synthetic data-based approaches for private on-device learning as an LLM preference optimization problem.

Contributions. In summary, our contributions are:

(1) We propose POPri, a novel method that casts private on-device learning under the synthetic data framework as an LLM preference optimization problem. Prior work in this space relied on PE, which uses client feedback exclusively to generate new prompts (Hou et al., 2024; Xie et al., 2024). We alter this feedback to instead provide client *preferences*, and subsequently exploit recent advances in preference optimization—namely, Direct Preference Optimization (DPO) (Rafailov et al., 2023).

(2) We demonstrate the utility of POPri on a new benchmark set of datasets (see contribution #3), as well as a dataset collected from PubMed (Yu et al., 2023; Xie et al., 2024). Across all datasets, POPri achieves the best downstream metrics. In Figure 1, on our bioRxiv dataset at privacy level $\epsilon = 1.0$, POPri outperforms PE-based algorithms by 2 percentage points, and closes the gap between fully private and non-private baselines by over 68%, compared to 52% for PE. It outperforms DP-FL-based methods even more. Additional experimental details, results, and ablations are provided in Section 5.

(3) We create and maintain LargeFedBench, an uncontaminated federated benchmark for LLMs, featuring client-separated data from: (1) congressional records in English-speaking countries, and (2)

abstracts from bioRxiv, collected starting in April 2023. To our knowledge, this is the first dataset with both (a) over 1,000 clients (congressional records contains 134k clients and bioRxiv contains 72k as of August 2024), and (b) regular updates, allowing researchers to filter data to avoid contaminated evaluations (Magar & Schwartz, 2022; Zhou et al., 2023; Yang et al., 2023; Roberts et al., 2023).

2 Problem Statement and Background

We consider a set $\mathcal{S}$ of clients, $\mathcal{S} = \{S_1, \ldots, S_n\}$, where $S_i = \{s_1^{(i)}, \ldots, s_{m_i}^{(i)}\}$ denotes the private text data of client $i \in [n]$, and m_i denotes the number of text samples held by client i. We consider the partial participation setting, where only a subset of clients can participate in communication with the server at any point in time (Kairouz et al., 2021a; McMahan et al., 2017a), reflecting practical private on-device learning deployments. We assume L clients participate in each round $t \leq T$ and denote this set $\mathcal{S}^t$. We do not assume an *a priori* upper bound on m_i. A central server aims to align a pre-trained downstream model Φ with private client data, producing an aligned model $\tilde{\Phi}$. The server may utilize a separate pre-trained public LLM Ψ in the process, assuming access to the weights of both Φ. and Ψ. However, it must adhere to two constraints: (1) client data cannot leave client devices, and (2) the final model $\tilde{\Phi}$ must protect user-level differential privacy (DP).

Neighboring datasets. We say two datasets $\mathcal{S}$ and $\mathcal{S}'$ are *neighboring* if they differ in at most one client's data (i.e., a user-level guarantee). That is, there exists an $i \in [n]$ such that for all $j \neq i$, $S_j = S'_j$.

User-level (distributed) differential privacy (DP). A randomized mechanism $\mathcal{M}$ is (ϵ, δ)-DP if, for any pair of neighboring datasets $\mathcal{S}$, $\mathcal{S}^{'}$ that differ by one sample and any possible output set E, it holds that $\Pr[\mathcal{M}(\mathcal{S}) \in E] \leq e^{\epsilon}\Pr[\mathcal{M}(\mathcal{S}^{'}) \in E] + \delta$. The post-processing property of a DP mechanism ensures that any data-independent transformation applied to its output preserves the same DP guarantees (Dwork, 2006; Dwork & Roth, 2014).

Goal The server seeks an algorithm to optimize the downstream next-word prediction performance of $\tilde{\Phi}$ on a test set of private client data, subject to an (ϵ, δ)-DP constraint.

Related work There are two main approaches for learning from private data in NLP tasks. The first are DP optimization-based approaches, where LLMs are fine-tuned using DP-SGD (Abadi et al., 2016a) on private data (Bommasani & Schofield, 2019; Kurakin et al., 2023; Charles et al., 2024). However, when client data cannot leave client devices, central servers cannot use this method. An alternative approach is to train models directly on client devices, using a method called DP-FL (McMahan et al., 2017a; Kairouz et al., 2021a). In DP-FL, (small) model weights are iteratively send to clients for on-device DP optimization. However, DP-FL cannot be done with large models like LLMs, which are too large to fit on client devices. The second type of approach for learning from private data are synthetic data based approaches. The idea is to create a synthetic version of the client data satisfying DP guarantees (Yue et al., 2023a; Mattern et al., 2022; Xie et al., 2024), which we can fine-tune models on. In the private on-device setting, Hou et al. (2024) show that fine-tuning a small model on DP synthetic text data on the server side can actually *outperform* DP-FL. Wu et al. (2024) show that pretraining an on-device model on DP synthetic text can improve DP-FL.

3 POPri

POPri (**P**reference **O**ptimization for **Pri**vate Client Data) is a natural reformulation of private on-device learning from synthetic data as an LLM preference optimization problem, which enables the use of powerful LLM alignment methods like DPO (Rafailov et al., 2023).

What client feedback should we collect? A major reason for the success of Private Evolution (PE) is the fact that it privately collects data structures that are low-dimensional (relative to model gradients). In PE, the server generates K synthetic data samples (Xie et al., 2024; Hou et al., 2024). Each client computes a histogram counting how often each of the private samples is closest to one of the K samples. The client returns a DP histogram by adding Gaussian noise.

POPri changes the feedback stage by asking the server to generate J samples from each of K prompts, which allows clients to build a preference dataset. Specifically, each client scores each synthetic sample by computing the average cosine distance between each synthetic sample and the private data. Using these client scores, the server can construct a "higher scoring response" and "lower scoring response" pair (a "preference pair") for each of the K prompts. We make use of this information as follows.

How should we use client feedback? Three natural candidates for using client feedback are: (1) *In-Context Learning.* We could use the highest-scoring samples as in-context examples for LLM Ψ, following the PE approach (Hou et al., 2024; Xie et al., 2024). However, in-context learning often underperforms compared to fine-tuning (Mosbach et al. (2023), Figure 1, Table 1). (2) *Supervised*

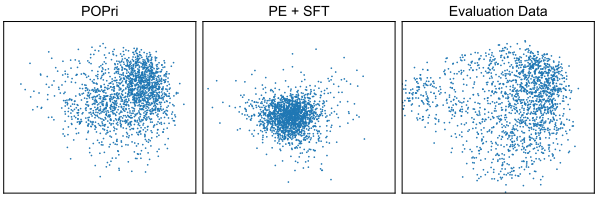

Figure 2: 2-PCA visualization of synthetic data from POPri and PE+SFT, and evaluation data. Naively fine-tuning with SFT on PE-generated synthetic data does not make best use of client feedback.

Fine-Tuning (SFT). One could fine-tune the LLM Ψ on the highest-scoring samples using next-word-prediction loss, similar to the SFT baseline evaluated by Ouyang et al. (2022) and Rafailov et al. (2023). However, the highest-scoring samples are not perfect, and SFT incorrectly treats them as ground truth (Figure 2 and Table 1). (3) *Preference Optimization (PO).* Methods like DPO (Rafailov et al., 2023) instead optimize the LLM to generate higher-scoring samples using preference pairs, leveraging low-dimensional scores from client feedback. POPri uses this approach, as we expect it to yield higher-quality synthetic data. We avoided RLHF (Ouyang et al., 2022) due to its high computational demands for training a reward model and did not choose IPO (Gheshlaghi Azar et al., 2024) based on an ablation in Appendix F.4.

3.1 POPRI ALGORITHM

Pseudocode can be found in Algorithm 1. Algorithmically new steps that differ from PE are in blue.

1. Initial sample population. We start with an initial set of samples Ω, which come from a publicly available source, either available on the internet or text generated by a publicly available LLM.

2. Synthetic sample generation. We create K prompts (prompt in Appendix C). For each of the K prompts, we generate J synthetic samples (by running the prompt independently J times). In total, the server generates $K \times J$ synthetic samples and sends them to every client in round t, $\mathcal{S}^t$.

3. Scoring the quality of the synthetic samples using private client feedback. Next, each client that received synthetic data score each synthetic sample. Specifically, each client calculates the average cosine similarity between each $K \times J$ synthetic sample and the entire client dataset (Algorithm 2). These similarities for every synthetic sample are arranged into a vector. We clip this vector to a norm of 1, which caps the contribution of each client (similar to how gradient updates are clipped per client in DP-FL (McMahan et al., 2017a)). This is done primarily for privacy reasons, as we will elaborate later. Clipping also ensures that the contribution of clients with large amounts of data does not overwhelm the contribution of clients with small amounts of data. We then add $\mathcal{N}(0, \sigma^2 I/L)$ (where I is the identity matrix of size $KJ \times KJ$) noise to the resulting vector to ensure DP (σ^2 controls the (ϵ, δ)). Finally, we aggregate scores via secure aggregation (Bonawitz et al., 2016), yielding a DP score for each synthetic sample that reflects its relevance to client data.

4. LLM Preference Optimization. Our key insight is that generating J synthetic samples from K prompts and scoring them with DP client feedback enables the creation of a preference dataset. For each prompt, we designate the highest-scoring sample as "chosen" and the ℓ-th highest as "rejected".

This resulting preference dataset can then be passed, along with the LLM Ψ, into the DPO preference optimization loss (Rafailov et al., 2023):

$$\min_{\Psi} \mathop{\mathbb{E}}_{\substack{x, y_\omega \\ y_r}} [-\log s(\tau \log(\frac{\Psi(y_\omega|x)}{\Psi(y_r|x)}) - \tau \log(\frac{\Psi_{\text{ref}}(y_\omega|x)}{\Psi_{\text{ref}}(y_r|x)}))]$$

where Ψ_{ref} is a fixed LLM checkpoint (we use a public one), τ controls Ψ's deviation from Ψ_{ref}, x is the prompt, y_ω and y_r are the chosen and rejected samples, $\Psi(y|x)$ is the generation probability, and s is the sigmoid function. The expectation is taken with respect to the empirical distribution (i.e. real samples). DPO loss fine-tunes Ψ to favor generating chosen samples over rejected ones. To reduce GPU memory use, we apply LoRA (Hu et al., 2021) with rank 4 and $\alpha = 8$ on all attention and projection matrices. After fine-tuning on K prompts and preference pairs, we return to step (2) to generate new synthetic data with the updated Ψ.

5. Synthetic data generation for downstream tasks. Using the final version of Ψ, we generate a large set of synthetic data $S_{syn,T+1}$ which is used to fine-tune Φ into $\tilde{\Phi}$. $\tilde{\Phi}$ is then sent to all the client devices, where they can perform inference without communicating information to the server.

Privacy guarantees. Because each client's vector is clipped to 1, and the only information revealed to the server (or any other party) is the aggregated vector, the sensitivity of the algorithm is 1. We add $\mathcal{N}(0, \sigma^2 I/L)$ noise to each client's vector, so the vector given to the server has noise $\mathcal{N}(0, \sigma^2 I)$, satisfying the Gaussian Mechanism with sensitivity 1. To calculate privacy, we can use a privacy accountant like `OPACUS.ACCOUNTANTS.ANALYSIS.RDP` (Yousefpour et al., 2021), and input T (number of rounds we run the algorithm), q (fraction of clients sampled per round), δ, and set σ to get the desired ϵ value.

4 LargeFedBench: A Federated Benchmark for LLM Evaluation

The most widely used federated learning text datasets, released by Reddi et al. (2020) include StackOverflow and Shakespeare text but present two challenges: (1) They pre-tokenize inputs in a non-invertible way, which prevents researchers from using custom tokenizers adopted by several LLMs. (2) They risk evaluation contamination, as state-of-the-art LLMs may have been trained on similar public datasets (Magar & Schwartz, 2022; Zhou et al., 2023; Yang et al., 2023; Roberts et al., 2023). To our knowledge, no benchmarks today have both production-level client numbers (at least 10,000) and prevent evaluation contamination (Ye et al., 2024).

We release **LargeFedBench**, a benchmark comprising two new datasets, Congressional Speeches and bioRxiv, for experiments over federated client data. These datasets (a) allow researchers to easily avoid contamination, and (b) provide enough distinct clients to simulate production settings. **Congressional Speeches ("Congress")**[1] contains 134k speeches or debates from US, UK, and Canadian transcripts. Each speech is treated as a client, with 64-token spans as samples. The **bioRxiv** dataset [2] includes 72k biology paper abstracts, each treated as a client dataset, with 64-token spans as samples. More details are in Appendix G.

Our datasets are updated every 6 months and sorted by date, so researchers can select datasets generated after their model's knowledge cutoff date. E.g., we use data from LargeFedBench published between the dates of April 2023 to August 2024 to avoid contamination with our LLM evaluation model, LLaMA-3-8B (AI@Meta, 2024)—which has a knowledge cutoff of March 2023.

5 Experiments

Datasets We evaluate POPri on the LargeFedBench datasets (**Congress** and **bioRxiv**), as well as a third **PubMed** dataset (Yu et al., 2023; Xie et al., 2024) used in the evaluation of Private Evolution (Aug-PE) (Xie et al., 2024). PubMed contains abstracts of medical papers published between August 1-7, 2023 (details in Appendix E.2.2).

[1] `https://huggingface.co/datasets/hazylavender/CongressionalDataset`
[2] `https://huggingface.co/datasets/hazylavender/biorxiv-abstract`

Table 1: Next-token prediction accuracy ($\%, \uparrow$) of different algorithms. The highest accuracy across all methods is in **bold**. All standard deviation error bars are less than 0.5.

Dataset	Method	Data Type	On-device Model	$\epsilon = \infty$	$\epsilon = 7$	$\epsilon = 1$	$\epsilon = 0$
bioRxiv	DP-FedAvg	Original	DistilGPT2	72.2	61.7	61.7	60.6
	DP-FTRL	Original			61.8	61.8	
	PE	Synthetic			66.2	66.3	
	PE + SFT	Synthetic			64.8	64.6	
	POPri (ours)	Synthetic			**68.6**	**68.6**	
Congress	DP-FedAvg	Original	DistilGPT2	74.5	69.2	69.2	68.4
	DP-FTRL	Original			69.1	69.1	
	PE	Synthetic			70.3	70.4	
	PE + SFT	Synthetic			70.0	70.2	
	POPri (ours)	Synthetic			**71.3**	**71.3**	
Dataset	**Method**	**Data Type**	**On-device Model**	$\epsilon = \infty$	$\epsilon = 4$	$\epsilon = 1$	
PubMed	PE	GPT-2-Large, Synthetic (2000)	$\text{BERT}_{\text{small}}$	47.6	27.9	27.2	
	PE	Llama-2-7b-chat-hf, Synthetic (2000)			—	27.5	
	PE	Opt-6.7b, Synthetic (2000)			—	27.9	
	POPri (ours)	Synthetic (2000)			**29.2**	**29.4**	

Models. We use LLaMA-3-8B (Grattafiori et al., 2024) as the LLM Ψ (knowledge cutoff: March 2023 AI@Meta (2024)) and 'all-MiniLM-L6-v2' for embedding-based semantic distance. DistilGPT2 (Sanh et al., 2019), with 82M parameters, serves as the on-device model. For synthetic text generation, the max sequence length is 64 for bioRxiv and Congressional Speeches, and 512 for PubMed. During training, we select the best validation-performing checkpoint for final evaluation.

Metrics. We primarily evaluate each method under next-token prediction accuracy of the final downstream on-device model $\tilde{\Phi}$. In some ablations we also measure the distance of the synthetic dataset to the private dataset using the Fréchet Inception Distance (FID) (Heusel et al., 2017).

Baselines. We compare POPri with baselines: (1) DP-FedAvg (McMahan et al., 2017b), (2) DP-FTRL (Kairouz et al., 2021a), and (3) Private Evolution methods (PrE-Text (Hou et al., 2024), Aug-PE (Xie et al., 2024)). DP-FedAvg and DP-FTRL fine-tune the downstream model Φ on client data, while Private Evolution generates synthetic data for fine-tuning. On PubMed, we compare Aug-PE using similarly sized models ($\sim$8B parameters), though their best results rely on GPT-3.5 (175B), limiting direct comparison. We also include fully private ($\epsilon = 0$) and fully non-private ($\epsilon = \infty$) baselines. The $\epsilon = 0$ baseline evaluates the public DistilGPT2 checkpoint without fine-tuning, while $\epsilon = \infty$ fine-tunes DistilGPT2 on the private training set. More details are in Appendices D and E.2.

All baselines use a privacy guarantee of (ϵ, δ)-DP where δ=3$\times 10^{-6}$ and ϵ=1 or ϵ=7 for each of the bioRxiv and Congressional Speeches datasets. For PubMed, we set $\delta = \frac{1}{N_{priv} \cdot \log(N_{priv})}$ (N_{priv} is the number of private samples) and ϵ=1 or ϵ=4 and fine-tune BERT_{small} for fair comparison to the results from Xie et al. (2024). Details for all baselines are in Appendix E.1.

5.1 Results

Table 1 shows next-token prediction accuracy for baseline models (DP-FedAvg, DP-FTRL, Private Evolution) and POPri, assuming full client participation for fair comparison. POPri outperforms all baselines and reduces the gap between fully private ($\epsilon = 0$) and fully non-private ($\epsilon = \infty$) learning by 48-69$\%$, compared to PE's 31-49$\%$. PE+SFT performs similarly or worse than PE. We find that accuracy is largely independent of ϵ, consistent with prior DP synthetic data work (Xie et al., 2024; Hou et al., 2024). POPri outperforms Aug-PE even with a 2000-sample budget. Note that POPri can generate many more samples than Aug-PE (Xie et al., 2024), because Aug-PE is limited to model API access and synthetic sample generation is costly.

We also perform comprehensive ablations on POPri. Our main findings are: (1) **Client participation:** As we vary client sampling or the number of clients that participate in each round, POPri consistently outperforms competing baselines. Moreover, it is less sensitive to number of clients than some FL-based methods (Figures 5 and 6). (2) **Number of rounds:** POPri does not monotonically improve as the number of rounds increases. Instead, some form of early stopping (in terms of rounds) is

necessary to obtain the best synthetic data quality and prevent overfitting (Figure 4). (3) **POPri Hyperparameters:** We demonstrate the effects of changing the preference optimization algorithm, the index of the "rejected" sample ℓ, and the temperature of LLM inference. These ablations are in Appendix F.

6 Conclusion

Private on-device learning is important when data is stored on edge devices with hardware, storage, and privacy constraints. We propose POPri, which recasts synthetic data-based approaches for private on-device learning as an LLM preference optimization problem. POPri makes several novel design choices in how it gathers and utilizes client feedback to generate DP synthetic data, which is used to finetune a downstream on-device model. POPri outperforms DP-FL and synthetic data baselines on the downstream next-word-prediction task, including on LargeFedBench, a new federated benchmark we have curated.

Our work is only a first step in learning private synthetic data with preference optimization, and many important questions remain. First and foremost, in POPri, it would be important to understand a more systematic method for selecting the ranked sample to be used as a "negative" sample in the preference optimization. Our current algorithm heuristically uses the 5th-ranked sample (out of 10) for the "negative" preference sample as a way to balance noisy preference feedback, which can reverse preference orderings. It would be interesting to understand if this heuristic could be learned (and adapted) in an online fashion, and/or if one could make use of robust preference optimization algorithms like that of Chowdhury et al. (2024). Moreover, it would be useful to understand how POPri (and any possible improvements on it) can be made robust to adversarial clients who provide adversarially incorrect preferences.

Acknowledgments

This work used Bridges-2 GPU (Brown et al., 2021; Buitrago & Nystrom, 2021) at the Pittsburgh Supercomputing Center through allocation CIS240135 and CIS240937 from the Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support (ACCESS) program, which is supported by National Science Foundation grants 2138259, 2138286, 2138307, 2137603, and 2138296 (Boerner et al., 2023). The authors acknowledge the National Artificial Intelligence Research Resource (NAIRR) Pilot, the AI and Big Data group at the Pittsburgh Supercomputing Center, and NCSA Delta GPU for contributing to this research result. GF was supported in part by NSF grants RINGS-214835 and CCF-2338772, C3.ai, Bosch, Intel, and the Sloan Foundation.

References

Martin Abadi, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, CCS'16. ACM, October 2016a. doi: 10.1145/2976749.2978318. URL http://dx.doi.org/10.1145/2976749.2978318.

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

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

AI@Meta. Llama 3 model card. 2024. URL https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md.

Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*, 2023.

Timothy J Boerner, Stephen Deems, Thomas R Furlani, Shelley L Knuth, and John Towns. Access: Advancing innovation: Nsf's advanced cyberinfrastructure coordination ecosystem: Services & support. In *Practice and Experience in Advanced Research Computing 2023: Computing for the Common Good*, pp. 173–176. 2023.

Wu S. Bommasani, R. and X. Schofield. Towards private synthetic text generation. *In NeurIPS 2019 Machine Learning with Guarantees Workshop*, 2019.

K. A. Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H. Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for federated learning on user-held data. In *NIPS Workshop on Private Multi-Party Machine Learning*, 2016. URL https://arxiv.org/abs/1611.04482.

Shawn T. Brown, Paola Buitrago, Edward Hanna, Sergiu Sanielevici, Robin Scibek, and Nicholas A. Nystrom. Bridges-2: A platform for rapidly-evolving and data intensive research. In *Practice and Experience in Advanced Research Computing 2021: Evolution Across All Dimensions*, PEARC '21, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450382922. doi: 10.1145/3437359.3465593. URL https://doi.org/10.1145/3437359.3465593.

P. A. Buitrago and N. A. Nystrom. Neocortex and bridges-2: A high performance ai+hpc ecosystem for science, discovery, and societal good. *Communications in computer and information science*, 1327, 2021. doi: 10.1007/978-3-030-68035-0_15. URL https://par.nsf.gov/biblio/10274872.

Zachary Charles, Nicole Mitchell, Krishna Pillutla, Michael Reneer, and Zachary Garrett. Towards federated foundation models: Scalable dataset pipelines for group-structured learning. *arXiv preprint arXiv:2307.09619*, 2023.

Zachary Charles, Arun Ganesh, Ryan McKenna, Hugh Brendan McMahan, Nicole Elyse Mitchell, Krishna Pillutla, and J Keith Rush. Fine-tuning large language models with user-level differential privacy. In *ICML 2024 Workshop on Theoretical Foundations of Foundation Models*, 2024.

Sayak Ray Chowdhury, Anush Kini, and Nagarajan Natarajan. Provably robust dpo: Aligning language models with noisy feedback. *arXiv preprint arXiv:2403.00409*, 2024.

Liam Collins, Shanshan Wu, Sewoong Oh, and Khe Chai Sim. Profit: Benchmarking personalization and robustness trade-off in federated prompt tuning. *arXiv preprint arXiv:2310.04627*, 2023.

Cynthia Dwork. Differential privacy. In *International colloquium on automata, languages, and programming*, pp. 1–12. Springer, 2006.

Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.*, 9(3–4):211–407, August 2014. ISSN 1551-305X. doi: 10.1561/0400000042. URL https://doi.org/10.1561/0400000042.

Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Remi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. A general theoretical paradigm to understand learning from human preferences. In Sanjoy Dasgupta, Stephan Mandt, and Yingzhen Li (eds.), *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pp. 4447–4455. PMLR, 02–04 May 2024. URL https://proceedings.mlr.press/v238/gheshlaghi-azar24a.html.

Aaron Grattafiori, Abhimanyu Dubey, and et al. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Shangmin Guo, Biao Zhang, Tianlin Liu, Tianqi Liu, Misha Khalman, Felipe Llinares, Alexandre Rame, Thomas Mesnard, Yao Zhao, Bilal Piot, Johan Ferret, and Mathieu Blondel. Direct Language Model Alignment from Online AI Feedback. *arXiv e-prints*, art. arXiv:2402.04792, February 2024. doi: 10.48550/arXiv.2402.04792.

Andrew Hard, Kanishka Rao, Rajiv Mathews, Swaroop Ramaswamy, Françoise Beaufays, Sean Augenstein, Hubert Eichner, Chloé Kiddon, and Daniel Ramage. Federated learning for mobile keyboard prediction, 2019. URL https://arxiv.org/abs/1811.03604.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *NeurIPS*, 30, 2017.

Charlie Hou, Akshat Shrivastava, Hongyuan Zhan, Rylan Conway, Trang Le, Adithya Sagar, Giulia Fanti, and Daniel Lazar. Pre-text: Training language models on private federated data in the age of llms, 2024. URL https://arxiv.org/abs/2406.02958.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

P. Kairouz, B. McMahan, S. Song, O. Thakkar, A. Thakurta, and Z. Xu. Practical and private (deep) learning without sampling or shuffling. In *ICML*, 2021a.

Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021b.

Alexey Kurakin, Natalia Ponomareva, Umar Syed, Liam MacDermed, and Andreas Terzis. Harnessing large-language models to generate private synthetic text. *arXiv preprint arXiv:2306.01684*, 2023.

Zinan Lin, Vyas Sekar, and Giulia Fanti. On the privacy properties of gan-generated samples. In *AISTATS*, pp. 1522–1530. PMLR, 2021.

Zinan Lin, Sivakanth Gopi, Janardhan Kulkarni, Harsha Nori, and Sergey Yekhanin. Differentially private synthetic data via foundation model apis 1: Images. *arXiv preprint arXiv:2305.15560*, 2023.

Inbal Magar and Roy Schwartz. Data contamination: From memorization to exploitation, 2022. URL https://arxiv.org/abs/2203.08242.

Sadhika Malladi, Tianyu Gao, Eshaan Nichani, Alex Damian, Jason D Lee, Danqi Chen, and Sanjeev Arora. Fine-tuning language models with just forward passes. *Advances in Neural Information Processing Systems*, 36:53038–53075, 2023.

Justus Mattern, Zhijing Jin, Benjamin Weggenmann, Bernhard Schoelkopf, and Mrinmaya Sachan. Differentially private language models for secure data sharing. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 4860–4873, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.323. URL https://aclanthology.org/2022.emnlp-main.323.

B. McMahan, D. Ramage, K. Talwar, and L. Zhang. Learning DP recurrent language models. 2017a.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-Efficient Learning of Deep Networks from Decentralized Data. In Aarti Singh and Jerry Zhu (eds.), *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pp. 1273–1282. PMLR, 20–22 Apr 2017b. URL https://proceedings.mlr.press/v54/mcmahan17a.html.

Marius Mosbach, Tiago Pimentel, Shauli Ravfogel, Dietrich Klakow, and Yanai Elazar. Few-shot fine-tuning vs. in-context learning: A fair comparison and evaluation. *arXiv preprint arXiv:2305.16938*, 2023.

John Nguyen, Kshitiz Malik, Hongyuan Zhan, Ashkan Yousefpour, Mike Rabbat, Mani Malek, and Dzmitry Huba. Federated learning with buffered asynchronous aggregation. In *International Conference on Artificial Intelligence and Statistics*, pp. 3581–3607. PMLR, 2022.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Matthias Paulik, Matt Seigel, Henry Mason, Dominic Telaar, Joris Kluivers, Rogier van Dalen, Chi Wai Lau, Luke Carlson, Filip Granqvist, Chris Vandevelde, Sudeep Agarwal, Julien Freudiger, Andrew Byde, Abhishek Bhowmick, Gaurav Kapoor, Si Beaumont, Áine Cahill, Dominic Hughes, Omid Javidbakht, Fei Dong, Rehan Rishi, and Stanley Hung. Federated evaluation and tuning for on-device personalization: System design applications, 2021. URL https://arxiv.org/abs/2102.08503.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 53728–53741. Curran Associates, Inc., 2023.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv e-prints*, 2019.

Sashank Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečnỳ, Sanjiv Kumar, and H Brendan McMahan. Adaptive federated optimization. *arXiv preprint arXiv:2003.00295*, 2020.

Manley Roberts, Himanshu Thakur, Christine Herlihy, Colin White, and Samuel Dooley. Data contamination through the lens of time, 2023. URL https://arxiv.org/abs/2310.10628.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Well-read students learn better: On the importance of pre-training compact models, 2019. URL https://arxiv.org/abs/1908.08962.

Shanshan Wu, Zheng Xu, Yanxiang Zhang, Yuanbo Zhang, and Daniel Ramage. Prompt public large language models to synthesize data for private on-device applications. *arXiv preprint arXiv:2404.04360*, 2024.

Chulin Xie, Zinan Lin, Arturs Backurs, Sivakanth Gopi, Da Yu, Huseyin A Inan, Harsha Nori, Haotian Jiang, Huishuai Zhang, Yin Tat Lee, Bo Li, and Sergey Yekhanin. Differentially private synthetic data via foundation model APIs 2: Text. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=LWD7upg1ob.

Liyang Xie, Kaixiang Lin, Shu Wang, Fei Wang, and Jiayu Zhou. Differentially private generative adversarial network. *arXiv preprint arXiv:1802.06739*, 2018.

Zheng Xu, Maxwell Collins, Yuxiao Wang, Liviu Panait, Sewoong Oh, Sean Augenstein, Ting Liu, Florian Schroff, and H Brendan McMahan. Learning to generate image embeddings with user-level differential privacy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7969–7980, 2023a.

Zheng Xu, Yanxiang Zhang, Galen Andrew, Christopher A. Choquette-Choo, Peter Kairouz, H. Brendan McMahan, Jesse Rosenstock, and Yuanbo Zhang. Federated learning of gboard language models with differential privacy, 2023b. URL https://arxiv.org/abs/2305.18465.

Shuo Yang, Wei-Lin Chiang, Lianmin Zheng, Joseph E. Gonzalez, and Ion Stoica. Rethinking benchmark and contamination for language models with rephrased samples, 2023. URL https://arxiv.org/abs/2311.04850.

Rui Ye, Rui Ge, Xinyu Zhu, Jingyi Chai, Yaxin Du, Yang Liu, Yanfeng Wang, and Siheng Chen. Fedllm-bench: Realistic benchmarks for federated learning of large language models. *arXiv preprint arXiv:2406.04845*, 2024.

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. Opacus: User-friendly differential privacy library in pytorch. In *NeurIPS 2021 Workshop Privacy in Machine Learning*, 2021. URL https://openreview.net/forum?id=EopKEYBoI-.

Da Yu, Arturs Backurs, Sivakanth Gopi, Huseyin Inan, Janardhan Kulkarni, Zinan Lin, Chulin Xie, Huishuai Zhang, and Wanrong Zhang. Training private and efficient language models with synthetic data from LLMs. In *Socially Responsible Language Modelling Research*, 2023. URL https://openreview.net/forum?id=FKwtKzglFb.

Da Yu, Peter Kairouz, Sewoong Oh, and Zheng Xu. Privacy-preserving instructions for aligning large language models, 2024. URL https://arxiv.org/abs/2402.13659.

Jinliang Yuan, Chen Yang, Dongqi Cai, Shihe Wang, Xin Yuan, Zeling Zhang, Xiang Li, Dingge Zhang, Hanzi Mei, Xianqing Jia, et al. Rethinking mobile ai ecosystem in the llm era. *arXiv preprint arXiv:2308.14363*, 2023.

Xiang Yue, Huseyin Inan, Xuechen Li, Girish Kumar, Julia McAnallen, Hoda Shajari, Huan Sun, David Levitan, and Robert Sim. Synthetic text generation with differential privacy: A simple and practical recipe. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1321–1342, Toronto, Canada, July 2023a. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.74. URL https://aclanthology.org/2023.acl-long.74.

Xiang Yue, Huseyin A. Inan, Xuechen Li, Girish Kumar, Julia McAnallen, Hoda Shajari, Huan Sun, David Levitan, and Robert Sim. Synthetic text generation with differential privacy: A simple and practical recipe, 2023b. URL https://arxiv.org/abs/2210.14348.

Liang Zhang, Bingcong Li, Kiran Koshy Thekumparampil, Sewoong Oh, and Niao He. Dpzero: Private fine-tuning of language models without backpropagation. In *Forty-first International Conference on Machine Learning*, 2024.

Kun Zhou, Yutao Zhu, Zhipeng Chen, Wentong Chen, Wayne Xin Zhao, Xu Chen, Yankai Lin, Ji-Rong Wen, and Jiawei Han. Don't make your llm an evaluation benchmark cheater, 2023. URL https://arxiv.org/abs/2311.01964.

A ALGORITHMIC DETAILS

Algorithm 1 `POPri`

Input: Clients private data $\{S_i\}_{i\in[n]}$, Number of rounds T, Number of generated samples N_{syn}, Noise multiplier σ, LLM Ψ, embedding model Γ, base prompt η, participating clients in each round $\mathcal{S}^t$, "rejected" index ℓ, initial sample set Ω, number of clients sampled L
Output: Synthetic data $S_{syn,T+1}$

All clients $i \in [n]$ embed private samples, $E_i = \Gamma(S_i)$
Server initializes LLM $\Psi_1 = \Psi$
for $t \leftarrow 1 \ldots T$ **do**
  Server:
  Initialize the response vector $R = \emptyset$
  for $k \leftarrow 1 \ldots K$ **do**
    Generate prompt $\eta_k = \Psi(\eta, \Omega)$,
    Generate J responses $R_{kj} = \Psi_t(\eta_k)$, $j \in [J]$
  end for
  Send embeddings $E_{syn,t} = \{\Gamma(R_{kj})\}_{k\in[K],j\in[J]}$ to all clients in $\mathcal{S}^t$

  Client $i \in \mathcal{S}^t$**:**
  $\text{Scores}_{i,t} \leftarrow \texttt{SIMILARITY}(E_{syn,t}, E_i)$
  Send $\text{Scores}_{i,t} + \mathcal{N}(0, \sigma^2 I/L)$ to Server

  Server:
  Securely aggregate DP client scores: $\text{Scores}_t = \frac{1}{n}\sum_i \text{Scores}_{i,t} + \mathcal{N}(0, \sigma^2 I)$
  Set $P[k, j]$ as the j-th highest score response for prompt η_k, according to Scores_t
  Initialize preference dataset $\mathcal{P}_t = \emptyset$
  for $k \leftarrow 1 \ldots K$ **do**
    Select positive synthetic sample: $\mathcal{P}_t[k, 1] = P_t[k, 1]$
    Select negative synthetic sample: $\mathcal{P}_t[k, 2] = P_t[k, \ell]$
  end for
  Fine-tune: $\Psi_{t+1} \leftarrow \text{DPO}(\Psi_t, \{\eta_k\}_{k\in[K]}, \mathcal{P}_t)$
end for
Server:
Output final synthetic data $S_{syn,T+1}$ from Ψ_T

Algorithm 2 `SIMILARITY`

Input: Embeddings of private client data E_i for $i \in \mathcal{S}^t$, embeddings of synthetic data E_{syn}, total synthetic samples $M = K \times J$
$\text{Scores} \leftarrow \mathbf{0}^M$
$\text{Scores}[j] = (1/|E_i|) \sum_{e_{pri}\in E_i} \frac{\langle e_{pri}, e_j\rangle}{\|e_{pri}\|\|e_j\|}$ for $e_j \in E_{syn}$
return Scores

B RELATED WORK

There are two main approaches for learning on private data.

DP optimization-based approaches In natural language processing (NLP) tasks with privacy constraints, DP optimization algorithms (e.g., DP-SGD Abadi et al. (2016b)) are often used to fine-tune massively pretrained LLMs on private data Bommasani & Schofield (2019); Kurakin et al. (2023); Charles et al. (2024). However, in settings where client data cannot leave client devices due to privacy concerns, central servers cannot conduct this private fine-tuning.

An alternative approach is to train models directly *on client devices*, using a server to coordinate information exchange between clients; in DP federated learning (DP-FL) McMahan et al. (2017a); Kairouz et al. (2021a), (small) model weights are iteratively sent to clients for on-device DP optimization. DP-FL has struggled to keep up with the growing size of LLMs; many LLMs cannot be

stored or trained on client devices Collins et al. (2023). Recent work explores how to train LLMs in the DP-FL framework. Proposed approaches include training only subsets of parameters Charles et al. (2023), as well as memory-efficient zero-order optimization Zhang et al. (2024); Malladi et al. (2023). However, these methods still require the storage of the entire model on-device, limiting their practicality.

Synthetic data-based approaches An alternative approach to DP optimization involves generating private synthetic data using LLMs, followed by directly fine-tuning downstream models. Server-side synthetic data generation bypasses client hardware limits, and DP's post-processing property allows reuse without extra privacy loss Yue et al. (2023a). In the centralized DP setting (where the server is trusted to gather all the data, as opposed to our private on-device setting), prior studies have shown that training downstream models on DP synthetic text achieves performance comparable to privately training on real data (Yue et al., 2023a; Mattern et al., 2022; Xie et al., 2024). In the **private on-device** setting, Hou et al. (2024) show that fine-tuning a small model on user-level DP synthetic text data on the server side can actually *outperform* DP-FL. Similarly, Wu et al. (2024) show that pretraining an FL model on private synthetic data can improve the final outcome of DP-FL.

One approach for generating synthetic text data is to fine-tune an LLM (with DP-SGD) on private data (Kurakin et al., 2023; Yu et al., 2024) and then using it for synthetic data generation. However, client hardware constraints render this approach infeasible on-device. Recent works have relied instead on privacy-aware prompt engineering Wu et al. (2024); Xie et al. (2018); Hou et al. (2024). An important framework by Lin et al. (2023) called **Private Evolution** (PE) is the basis for several competitive DP synthetic text algorithms, including Aug-PE Xie et al. (2024) and PrE-Text Hou et al. (2024). Roughly, these algorithms use the public LLM Ψ to generate synthetic data, score each synthetic data according to its closeness to the client data, and discard synthetic data with low scores. The surviving synthetic data are used as in-context examples for Ψ to generate synthetic data. Private Evolution may sacrifice data quality in two ways: First, it uses in-context learning, which is often less effective than fine-tuning (Mosbach et al., 2023). Second, discarding low-score synthetic data may lose useful information (Ouyang et al., 2022). We address both by turning the DP synthetic generation problem into an LLM preference optimization problem.

C Implementation Details of POPri

C.1 Model and Hyperparameters

We choose LLaMA-3-8B as the data generator in POPri and we fine-tune it iteratively during the course of the algorithm. To fine-tune the LLaMA-3-8B model, we use LoRA fine-tuning with rank 4, α = 8, applied to all the projection matrices in LLaMA-3-8B. We adapt the AdamW optimizer with a cosine learning rate scheduler with the learning rate ranging from $3 \cdot 10^{-7}$ to $8 \cdot 10^{-7}$. In the Congress and bioRxiv evaluations, the sample set Ω is a subset of the c4 dataset (Raffel et al., 2019), which is a large scale dataset from 2019, which we use for fair comparison with Private Evolution (PrE-Text), though we do not know their exact initial sample set because they did not release it. For the PubMed evaluation, the sample set Ω is a set of 2000 samples generated using the PubMed generation prompt in Table 16 of the Aug-PE paper, generated by LLaMA-3-8B-Instruct (which has a knowledge cutoff of March 2023), for comparison with Aug-PE (Xie et al., 2024). For each iteration, we fine-tune the models for 2 epochs and select the best checkpoint with the lowest FID score relative to the validation dataset. This checkpoint is used for synthetic data generation and as the starting point for the next iteration. The batch size is set to 24.

In each round we generate 18000 synthetic data samples for the clients to evaluate. This is accomplished with 1800 prompts, each generating 10 samples for clients to rank. We select the 1st and 5th ranked sample for a given prompt for the "selected" and "rejected" data samples in the DPO preference dataset. We describe the experiments regarding which rank to use for constructing the preference dataset in detail in Appendix Section F.5. To test the scaling relation with the number of clients per round and the total number of clients participating in the training, we set up the parameters and privacy budget shown in Table 2. The 'all-MiniLM-L6-v2' sentence transformer model is used as the embedding model in POPri. We note that we adopt "sentence-t5-base" sentence transformer for PubMed during the step of fine-tuning BERT_{small}, which follows the setting in AUG-PE. We ensure POPri follows privacy guarantee of (ϵ, δ)-DP = $(1, 3 \times 10^{-6})$ or $(7, 3 \times 10^{-6})$ for both the

List of 6 diverse original text samples:

Original Text Sample 1
The observations showed that the object is four million times more massive than the sun and is the size of one astronomical unit (AU), a span equal to Earth's distance from the sun. Sgr A* has a mass density at least a trillion times greater than any known cosmic object.

Original Text Sample 2
In response to the general question, they need to study self-protection away from their marital baggage. They need to learn about home security, mobile security, the nature of crime, de-escalation, the law, escape tactics, awareness, and on and on. When it

Original Text Sample 3
Under the Patriot Act of 2001, the government significantly expanded its authority in regards to electronic surveillance (Henderson, 2002). One of the chief complaints is that the government can investigate anything that is considered "significant." The problem here is that there is

Original Text Sample 4
The life history advance program shall be funded from any of the following: monies provided by the general fund; amounts in the presidential family partnership fund; or monies provided by the revolving fund.

Original Text Sample 5
As you meet with employers this summer, get in touch with the team....

Figure 3: The synthetic data generation prompt for POPri. The black text marks the input prompt, and the brown text after "Original Text Sample 4" is generated. The generated text between "Original Text Sample 4" and "Original Text Sample 5" is collected and used as a synthetic sample.

bioRxiv and the Congressional Speeches datasets and run with 20 iterations for DP-FedAv, DP-FTRL, PrE-Text for comparison. For AUG-PE, we set (ϵ, δ)-DP = (1, 2.72×10^{-6}) or (4, 2.72×10^{-6}). PubMed experiments are run with 10 iterations.

In terms of models for downstream tasks:

- For BioRxiv & Congressional Speeches, we fine-tuned the pre-trained DistillGPT2 for next-token prediction. We set the max sequence length as 64, number of generated synthetic data as 1,000,000, the batch size as 160, the learning rate as $2e^{-4}$, and the number of epochs as 80.
- For PubMed, to compare with (Yue et al., 2023b), we follow their procedure to leverage pre-trained BERT_{small} Turc et al. (2019). We set the max sequence length as 512, number of generated synthetic data as 2000, batch size as 32, learning rate as 3e-4, the weight decay as 0.01, and the number of epochs as 10. To compare with Xie et al. (2024), we set up the (ϵ, δ)-DP value and hypterparameter according to their choice. For example, they set $\delta = \frac{1}{N_{priv} \cdot \log(N_{priv})}$ following Yue et al. (2023b). To achieve δ = {1,4}, we use noise multiplier σ = {13.7, 3.87} for 10 iterations under DP on all PubMed data. Note that our noise multiplier values are slightly different than Xie et al. (2024) due to different methods for calculating differential privacy.

C.2 Prompt Design

To compare with other data generator methods, we adopt the prompts used in the baseline models against which we compare. We generate the synthetic data using an approach similar to that in PrE-Text Hou et al. (2024). Figure 3 shows an example of the prompt we use for prompting Llama-3B for generating synthetic data. For PubMed, while running POPri, we still adopt the prompt shown in Figure 3 but reduce the number of examples to two in order to accommodate longer sequence lengths.

D Implementation Details of Baseline Models

In this section we provide implementation details for the baseline algorithms. We use two DP-FL baselines: DP-FedAvg and DP-FTRL. For the PE baseline, we implement PrE-Text Hou et al. (2024)

for the evaluations on the bioRxiv and Congressional Speeches datasets. Because the PrE-Text evaluation is focused on datasets with samples with max sequence length of 64 and the PubMed dataset has samples with longer sequence lengths, for the PE baselines on the PubMed dataset we directly compare against the Aug-PE results from Xie et al. (2024).

D.1 DP-FedAvg

We employ the FedAvg federated optimization algorithm McMahan et al. (2017a) to fully fine-tune DistilGPT2, avoiding linear probing due to its poor performance in DP language models Lin et al. (2021). Our training configuration includes a batch size of 2, a sequence length of 64, 20 communication rounds, and either full or partial client participation. For differential privacy (DP), we utilize secure aggregation Bonawitz et al. (2016) and introduce Gaussian noise McMahan et al. (2017a). We evaluate the model using next-token prediction accuracy across various numbers of training epochs on the clients. We tune the learning rate within the range [0.01, 0.06] and the clipping threshold between [0.01, 0.4], selecting the model with the best performance on the evaluation set for reporting. The noise is scaled to ensure a privacy guarantee of (ϵ, δ)-DP where $\delta = 3{\cdot}10^{-6}$ and ϵ = {1,7}, representing two distinct privacy regimes. The noise multipliers are σ = {19.3, 3.35} when considering all the data, and the settings for partial participation experiments are shown in Table 2.

D.2 DP-FTRL

We also use the DP variant of Follow-The-Regularized-Leader (DP-FTRL) algorithm McMahan et al. (2017a), which shows amplified results comparing to FedAvg without using privacy amplification, to fully fine-tune DistilGPT2. The hyperparameter settings are similar to DP-FedAvg other than the noise multipliers. The noise multipliers are σ = {19.5, 3.35} when considering all the data, and the settings for partial participation experiments are shown in Table 2.

D.3 PrE-Text

We follow similar settings as Hou et al. (2024) with some modifications. The privacy budget is similar to DP-FedAvg and POPri, with a privacy guarantee of (ϵ, δ)-DP where $\delta = 3{\cdot}10^{-6}$ and ϵ = {1,7} with σ = {19.3, 3.35} for full participation and partial participation in Table 2. We set the thresholds H = 0.1626, T = 20, and N_{syn} = 1024. We adopt the "all-MiniLM-L6-v2" sentence transformer model for text embedding generation.

E Experimental Details

E.1 Privacy Accounting

The precise privacy settings we use and their corresponding ϵ values, as calculated by their corresponding privacy budget computation methods, are reported in Table 2. DP-FedAvg (McMahan et al., 2017a) and Private Evolution (PrE-Text) (Hou et al., 2024) both use the Gaussian mechanism, and thus use similar computations. In both cases, we use the privacy accountant of the Opacus library Yousefpour et al. (2021). For DP-FedAvg, we calculate privacy by inputting the number of rounds, the client sampling ratio, setting the noise multiplier to be the product of σ and the clipping threshold, choosing a $\delta \ll 1/|\mathcal{S}|$, and setting σ for the desired ϵ. Private Evolution (PrE-Text) (Hou et al., 2024) also uses the Gaussian mechanism, so we use the same accounting except the noise multiplier is the product of σ and the maximum number of samples per client. For DP-FTRL, we follow the privacy accounting methods from their implementation. For Private Evolution (Aug-PE) (Xie et al., 2024), we report their reported ϵ directly.

E.2 Evaluation Details for Different Datasets

E.2.1 LargeFedBench Evaluation

For the bioRxiv and Congressional Speeches datasets, we use the PrE-Text version of Private Evolution because the PrE-Text evaluation focused on datasets with samples with max sequence length of 64.

Table 2: Experiment privacy budget settings.

Total # of clients	# of clients per round	σ_1 [a], $\epsilon = 7$	σ_1 [a], $\epsilon = 1$	σ_2 [b], $\epsilon = 7$	σ_2 [b], $\epsilon = 1$
10000	500	0.67	1.6	7.5	43
10000	1000	0.82	2.5	6.7	39
10000	2000	1.09	4.3	5.8	34
10000	2500	1.23	5.2	5.8	34
10000	5000	1.92	9.9	4.8	28
10000	7500	2.63	14.7	3.35	19.5
10000	10000	3.35	19.3	3.35	19.5
1000	1000	3.35	19.3	3.35	19.5
2000	1000	1.92	9.9	4.8	28
4000	1000	1.23	5.2	5.8	34
17000	1000	0.7	1.8	7.5	44
72000	1000	0.52	1.14	8.9	52
133000	1000	0.475	1.05	9.5	55
72000	72000	3.35	19.3	3.35	19.5
133000	133000	3.35	19.3	3.35	19.5

[a] For DP-FedAvg, PrE-Text, POPri.
[b] For DP-FTRL

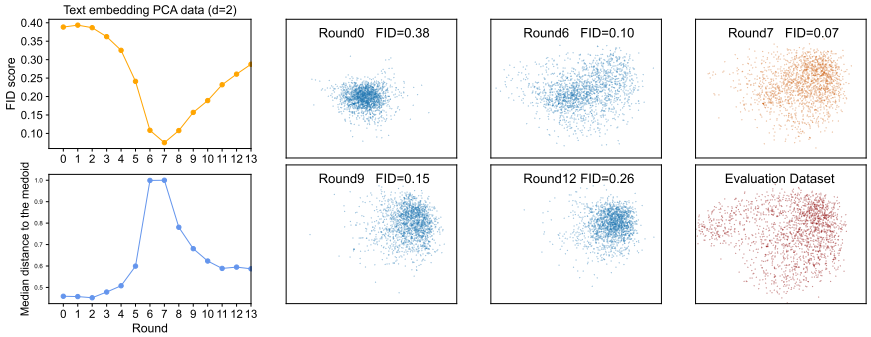

Figure 4: PCA visualization of POPri synthetic data embeddings over rounds. Right (6) Panels: PCA-2 plots of synthetic and evaluation data from the best checkpoint each round for 20 iterations. The orange (round 7) and maroon clouds represent the lowest FID score and validation set, respectively. Top Left: FID score vs. rounds. Bottom Left: Median distance to the medoid vs. rounds. Excessive rounds lead to overfitting.

E.2.2 PubMed Evaluation

For PubMed, our Private Evolution baseline compares to Aug-PE, which has already been evaluated on PubMed Xie et al. (2024). Note that PubMed was used by Xie et al. (2024) to evaluate central DP algorithms. In the central DP setting, there are no clients; all private data is held at the server and the goal is to release a model with DP guarantees. The notion of neighboring dataset in central DP is a centrally held dataset that is the same except for a single data sample. To compare our algorithm directly with results reported for Private Evolution (Aug-PE) (Xie et al., 2024), we replicate the central DP setting for this dataset by having one PubMed abstract per client and sampling all clients every iteration (or "round", in our case). We do not compare directly with the results reported in the PrE-Text paper (Hou et al., 2024) because they did not release the precise datasets used.

F Ablation Studies

F.1 Data Distribution Evolution

Synthetic datasets are often generated using a different language model than the one being aligned (Guo et al., 2024), making alignment off-policy as the model evolves. This impacts synthetic data quality, with FID scores initially improving but then worsening. Figure 4 shows PCA embeddings across iterations, where data distribution shifts from clustered to true-like, then back—likely due to overfitting. Early stopping can mitigate this.

Table 3: Ablation results including varying which client ranked-data is chosen as the 'rejected' sample for DPO, varying the temperature for the text synthesis process, and varying the preference optimization method.

Rank		Temperature		Alignment Algorithm	
Rank	Accuracy (%)	Temperature	Accuracy (%)	Algorithm	Accuracy (%)
3th	68.6	0.5	68.1	DPO	68.6
5th	68.6	1.0	68.6	IPO	67.1
7th	67.7	2.0	68.4		
10th	66.4				

F.2 Client selection strategies

Each round, a fixed number of clients is randomly subsampled for feedback. Figure 5 shows next-token prediction accuracy (%) across varying clients per round and total clients. POPri consistently outperforms baselines and remains robust to changes in client count, unlike most baselines. Figure 6 shows the effect of changing the number of clients per round and the number of total clients when $\epsilon = 7.0$. We find that POPri outperforms the baseline across the board. We also find that POPri is significantly less sensitive to the fraction of clients used per round than the baseline methods. This makes POPri especially useful in settings in which baseline performance may suffer, such as very high client participation regimes.

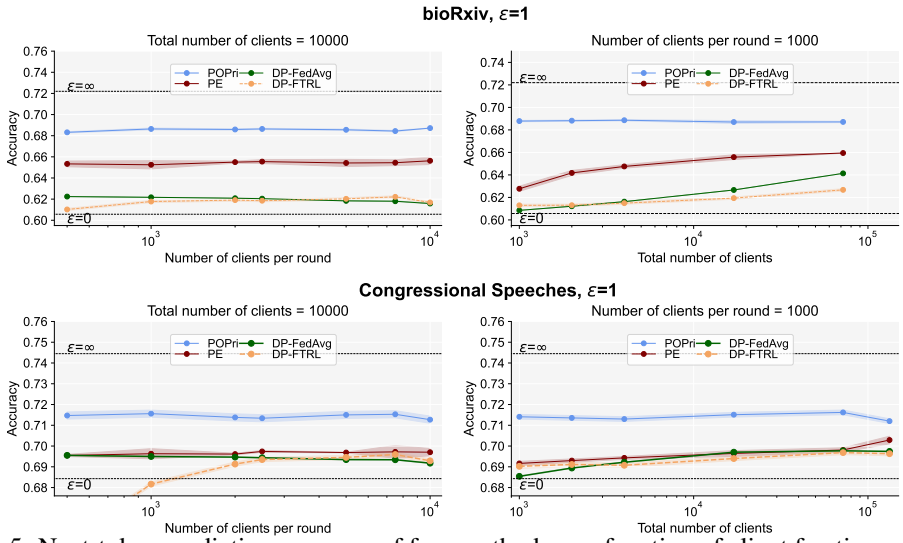

Figure 5: Next-token prediction accuracy of four methods as a function of client fraction per round. POPri improves performance and reduces sensitivity to client fraction. Left panels vary clients per round with the same total, right panels vary total clients with the same per-round count. Top and bottom panels use different datasets, with all methods having privacy budget (ϵ, δ)-DP = (1, 3×10^{-6}).

F.3 Temperature

Temperature is a key parameter for controlling the diversity of LLM-generated outputs. Increasing the temperature encourages the model to produce less frequent tokens, enhancing diversity. Here we explore the effects of changing temperature in the POPri process and list the results in Table 3. Low temperatures leads to clustering of the text embedding in some regions which does not represent the over all data distribution. However, setting the temperature too high can lead to overly random and potentially incoherent results. Therefore they both lead to lower model accuracy. We therefore choose temperature = 1.0 as our default setting.

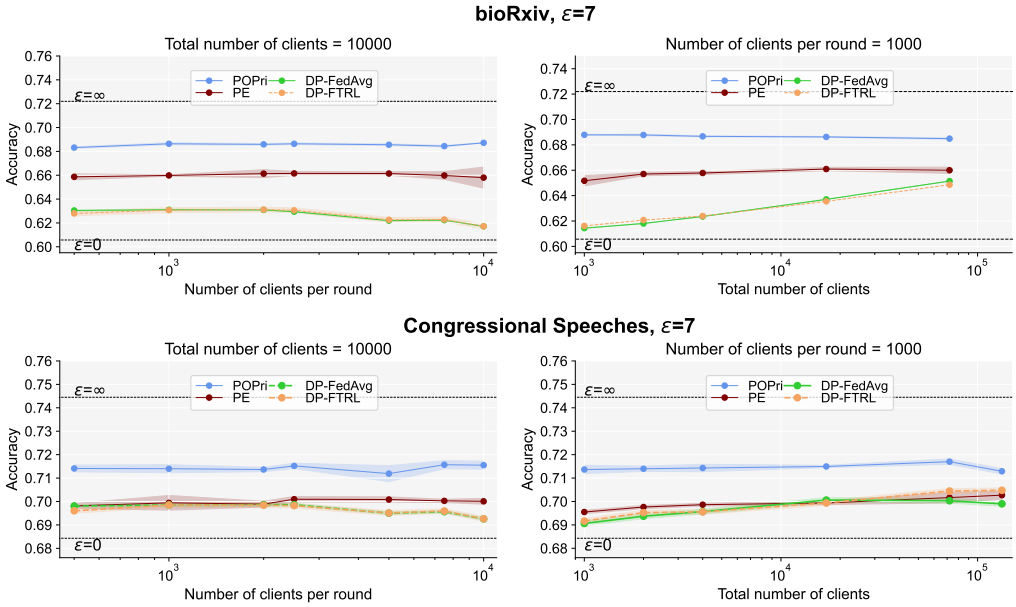

Figure 6: Next-token prediction accuracy (%) of four methods with privacy budget (ϵ, δ)-DP = $(7, 3 \times 10^{-6})$ for different number of clients per round with the same total number of clients (left panel) and different total number of clients with the same number of clients per round (right panel). Top two panels are generated with Biorxiv data, and the bottom two panels are generated with the Congressional Speeches dataset.

F.4 Alignment methods

As shown in Fig 4, over-fitting causes data clustering that misrepresents the true distribution. We compare two alignment methods: DPO and IPO (Gheshlaghi Azar et al., 2024), which IPO is designed to mitigate DPO's overfitting to preference data. In this section we explore IPO's performance in our setting. According to Gheshlaghi Azar et al. (2024), IPO may help training by alleviating over-fitting, which is a common problem for the DPO algorithm and which affects POPri as well. We show the comparison of next-token prediction accuracy reported by running DPO and IPO algorithm in Table 3. In our case, IPO does not seem to address the overfitting issues and results in worse performance. We therefore choose DPO as our alignment method for finetuning the LLM.

F.5 Rejected sample selection

Unlike vanilla DPO, we select the "chosen" and "rejected" pair from J samples for each of K prompts. The highest-scoring sample is always the "chosen," but the "rejected" sample could vary. In detail, we construct the DPO preference data via client feedback by generating ten samples from the same prompt and then picking the "selected" and the "rejected" samples. The samples with the highest scores among the ten examples are picked as the "selected" sample in the DPO preference dataset. We experiment on which rank should utilized as the "rejected" sample in the DPO preference dataset. In Table 3 we show the results of varying which rank is selected for the "rejected" sample. Perhaps surprisingly we find that the 10th rank is not favored. In Fig 7 we further explore the effects by examining the "rejected" and "selected" sample FID scores as a function of round. In the left panel where the "selected" sample FID values are shown, their magnitude and trends behave similarly before they reach the best results (marked by colored dashed vertical lines). For the "rejected" sample FID shown in the right panel, the 5th rank "rejected" samples yield the lowest FID score and therefore smaller gap between the preference sample pairs. However, we also find that smaller rank does not always yield better results. This may result from the boundary between the "rejected" and "selected" samples becoming undistinguishable for rank $<$ 5th due to DP noise. We therefore select 5th rank samples as our "rejected" DPO preference samples.

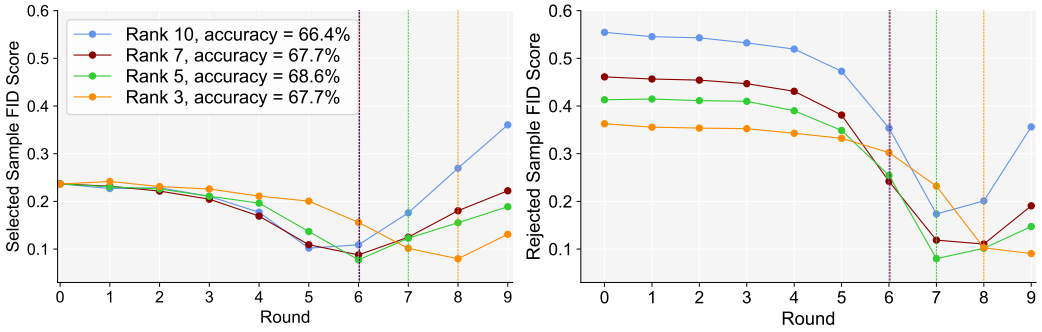

Figure 7: Ablation study for selecting rejected sample in the preference data. Here we generate 10 samples for each prompt and select Nth ranked data as the rejected sample, where N is 3, 5, 7, or 10. The vertical lines indicate the round at which the best next-word-prediction accuracy was achieved for each choice of rank. Note that the model that produces the lowest overall FID (not the lowest selected sample FID or the lowest rejected sample FID) is the best synthetic data generation model, since on the final round all generated samples are utilized to form the synthetic dataset. We hypothesize that round 7 corresponds to the highest accuracy for the rank 5 model because after that point, the selected sample FID is higher than the rejected sample FID, which would mean the preference dataset has become mis-aligned with the objective of generating good synthetic data.

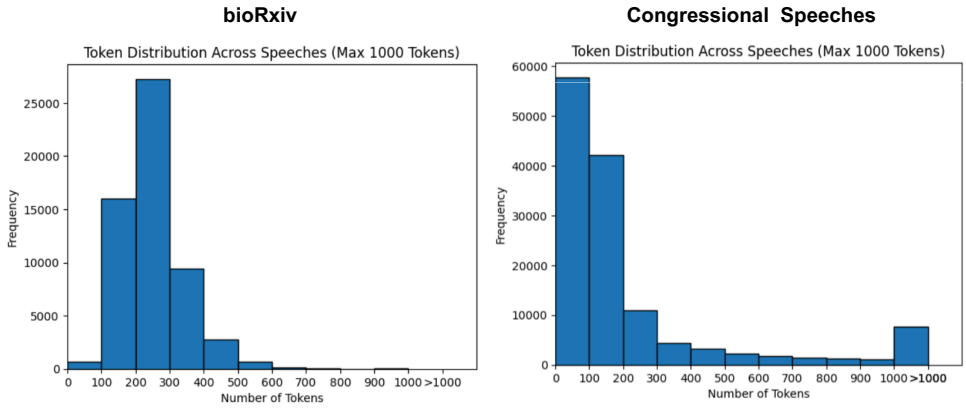

Figure 8: The distribution of how many tokens are in each client's dataset for the bioRxiv and Congressional Speeches datasets.

G DATASETS

bioRxiv. This dataset consists of abstracts from bioRxiv papers with appropriate copyright permission from April 2023 to August 2024. This was done by using the bioRxiv public API to retrieve the abstracts of the paper with permitted licenses (i.e. 'CC BY NC ND', 'CC BY ND', 'CC BY NC', 'CC BY', 'CC0'). This dataset consists of 72k abstracts (clients), each of which we split into chunks of 64 tokens to form samples.

Congressional Speeches. This dataset consists of speeches from US, UK and Canada congressional/parliamentary transcripts from April 2023 to August 2024. All speeches are published under a permissive license which allows for third-party use (as detailed in the dataset cards). There are 134k speeches (clients) in total, and 1930 unique speakers. We collected this dataset by using public APIs to retrieve data from each country's official congressional/parliamentary library website. Then we sanitized the data by removing (1) boilerplate procedural language, (2) sentences with more than 30% of the characters not being letters, and (3) some written notation that does not correspond to spoken words. We split each speech into chunks of 64 tokens each. We believe that this dataset is a major contribution because spoken language may be more resistant to contamination (especially for the UK

Table 4: Dataset details.

Dataset	# Train Samples	# Validation Samples	# Test Samples	Max Sequence Length	Average # of samples per client
bioRxiv	72000	2000	1584	64	6.6 ± 2.6
Congressional Speeches	133000	4200	1547	64	5.0 ± 16.3
PubMed	75316	14423	4453	512	1

and Canada parliamentary debates). Because they are more conversational and have a large degree of improvisation (many debates are off-the-cuff), they are less likely to be generated by LLMs.

We are committed to update the dataset periodically with the latest data to allow future researchers to test their algorithms or ideas against an uncontaminated dataset.