Differentially-Private Meta-Learning

Jeffrey Li CMU
Mikhail Khodak CMU
Sebastian Caldas CMU
Ameet Talwalkar CMU, Determined AI

Abstract

Parameter-transfer is a well-known and versatile approach for meta-learning, with applications including few-shot learning, federated learning, and reinforcement learning. However, parameter-transfer algorithms often require sharing models that have been trained on the samples from specific tasks, thus leaving the task-owners susceptible to breaches of privacy. We conduct the first formal study of privacy in this setting and formalize the notion of *task-global differential privacy as a practical relaxation of more commonly studied threat models*. We then propose a new differentially private algorithm for gradient-based parameter transfer that not only satisfies this privacy requirement but also retains provable transfer learning guarantees in convex settings. Empirically, we apply our analysis to the problem of federated learning with personalization and show that allowing the relaxation to task-global privacy from the more commonly studied notion of *local privacy* leads to dramatically increased performance in recurrent neural language modeling.

1 Introduction

The field of *meta-learning* offers promising directions for improving the performance and adaptability of machine learning methods. However, the collaborative nature of meta-learning, in which task-specific information is sent to and used by a *meta-learner*, also introduces inherent data privacy risks. In this work, we focus on a popular and flexible meta-learning approach, *parameter transfer* via *gradient-based meta-learning* (GBML). This set of methods, which includes well-known algorithms such as MAML [15] and Reptile [26], learns a common initialization ϕ over a set of tasks $t = 1, \ldots, T$ such that a high-performance model can be learned in only a few gradient-steps on new tasks. At each step, the meta-learner obtains feedback on the current ϕ in the form of task-specific updates $\hat{\theta}_t$.

Meanwhile, in many settings it is crucial to ensure that sensitive information in each task-specific dataset stays private. Examples of this include learning models for next-word prediction on cell phone data [25], clinical predictions using hospital records [31], and fraud detectors for competing credit card companies [29]. In such cases, each data-owner can benefit from information learned from other tasks, but each also desires, or is legally required, to keep their raw data private. While parameter transfer algorithms can move towards this goal by calculating $\hat{\theta}_t$'s locally, this provision alone is not fail-safe. Many works have shown in the single-task setting that an adversary with only access to a machine learning model can learn detailed information about the training set [27, 16, 9]. As such the meta-learner or any future task can potentially recover data from a previous task.

Despite these serious risks, privacy-preserving meta-learning has remained largely an unstudied problem. Our work aims to address this issue through the lens of *differential privacy* (DP) [14], a well-established definition of privacy with rich theoretical guarantees. Crucially, though there are various degrees of DP one could consider in the meta-learning setting, we balance the trade-off between privacy and model utility by formalizing and focusing on a notion that we call *task-global* DP. This provides a guarantee for each task-owner no single training example will be inferrable by *any* downstream agent. It also allows us to use the framework of Khodak et al. [21] to provide a DP GBML algorithm that enjoys provable learning guarantees in convex settings.

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

Finally, we show an application of our work by drawing connections to federated learning (FL). While standard FL methods, such as FedAvg [24], have inspired many works also concerning DP in a multi-user setup [2, 6, 17, 25, 30], we are the first to consider task-global DP as a useful variation on standard DP settings. Also, these works fundamentally differ in that they do not consider a task-based notion of learnability as they learn one global model. However, a federated setting involving per-user personalization [11, 28] is a natural meta-learning application. Our contributions are thus:

1. We are the first to outline the different possible DP notions for meta-learning and formalize a variant called *task-global* DP, showing that it adds a useful option for trading privacy and accuracy.
2. We propose the first DP GBML algorithm, which we construct to satisfy this privacy setting.
3. While our DP guarantees hold generally, we also prove learning-theoretic results in convex settings. These scale with task-similarity, as measured by closeness of optimal task-parameters [12, 22].
4. We show that our algorithm and theory naturally carries over to FL with personalization. Compared to previous notions of privacy considered in works for FL [2, 13, 17, 25], we are, to the best of our knowledge, the first to simultaneously provide both privacy and learning guarantees.
5. Empirically, we demonstrate that our proposed privacy setting allows for strong performance on non-convex federated language modeling tasks. We achieve nearly the performance of non-private models and significantly improve upon the performance of models trained with local-DP guarantees. Our setting reasonably relaxes this latter notion but can achieve roughly 2.0–3.7 times the accuracy on a modified version of the Shakespeare dataset [8] and 2.6 – 4.7 times the accuracy on a modified version of Wiki-3029 [3] across various privacy budgets

In the remainder of this paper, we outline the threat model and notion of DP that we provide in Section 2, sketch our theoretical results for this setting in Section 3, and summarize our experiments in Section 4. We defer more detailed discussions of related work and theory to Appendices A and C.

2 Privacy in a Meta-Learning Context

In this section, we describe the various threat models that arise in the meta-learning setup and present the different DP notions that can be achieved. However, we first describe the meta-learning setting that we will study. We consider tasks $t = 1, \dots, T$, each with its own training set $D_t = \{z_{t,i}\}_{i=1}^{m_t}$, where $z_{t,i} \in \mathcal{X} \times \mathcal{Y}$. The goal on each task is to learn a function $f_{\hat{\theta}_t} : \mathcal{X} \to \mathcal{Y}$ parameterized by $\hat{\theta}_t \in \Theta \subset \mathbb{R}^d$ that performs "well", generally in the sense that it has low within-task risk. The meta-learner's goal is to learn an initialization $\phi \in \Theta$ that leads to a well-performing $\hat{\theta}_t$ within-task. In GBML this ϕ is learned via an iterative process that alternates between: (1) a within-task procedure where a batch of task-owners B receives the current ϕ and each $t \in B$ uses ϕ as an initialization for running a within-task optimization procedure, obtaining $\hat{\theta}_t$; (2) a meta-level procedure where the meta-learner receives these model updates $\{\hat{\theta}_t\}_{t \in B}$ and aggregates them to update ϕ.

As in any privacy endeavor, a key specification must be made in terms of what threat model is being considered. In particular, it must be specified both (1) who the potential adversaries are and (2) what information needs to be protected. In meta-learning, natural options exist for each consideration.

Potential adversaries. For a task-owner, adversaries may be either solely recipients of ϕ (i.e. other task-owners) or recipients of either ϕ or $\hat{\theta}_t$ (i.e. also the meta-learner). In the latter case, we consider only a honest-but-curious meta-learner, who does not deviate from the agreed upon algorithm but may try to make inferences based on the information it receives. In both cases, concern is placed not only about the intentions of these other participants, but their own security.

Data to be protected. Protection can be given to either information in single records $z_{t,i}$ or entire datasets D_t simultaneously. This distinction between *record-level* and *task-level* privacy is important depending contextually on what kind of sensitive information exists. For instance, multiple $z_{t,i}$ within D_t may reveal the same secret (e.g., a cell-phone user has sent their SSN multiple times), or the entire distribution of D_t could be sensitive (e.g. a user has sent all messages in a foreign language). In these cases, record-level privacy may not be sufficient. However, given that privacy and utility are often at odds, we often seek the weakest notion of privacy needed in order to best preserve utility.

In related work, focus has primarily been placed on task-level protections. However, these works usually fall into two extremes, either obtaining strong learning but having to trust a central meta-learner, [25, 17] or trusting nobody but also obtaining low performance [6]. In response, we try to bridge the gap between these threat models by considering a model that makes a relaxation from task-level to record-level privacy but retains protections for each task-owner against *all* other parties. In practical situations, while task-level guarantees are strictly stronger, they may also be

Algorithm 1: Online version of our (ε, δ)-meta-private parameter-transfer algorithm.

Meta-learner picks first meta-initialization $\phi_1 \in \Theta$.
for task $t \in [T]$ **do**
  Meta-learner sends meta-initialization ϕ_t to task t.
  Task-learner runs OGD starting from $\theta_{t,1} = \phi_t$ on losses $\{\ell_{t,i}\}_{i=1}^m$, suffering regret $\sum_{i=1}^m \ell_{t,i}(\theta_{t,i}) - \min_{\theta \in \Theta} \sum_{i=1}^m \ell_{t,i}(\theta)$.
  Task-learner t runs (ε, δ)-DP descent algorithm on losses $\{\ell_{t,i}\}_{i=1}^m$ to get $\hat{\theta}_t$.
  Task-learner sends $\hat{\theta}_t$ to meta-learner.
  Meta-learner constructs loss $\ell_t(\phi) = \frac{1}{2}\|\hat{\theta}_t - \phi_t\|_2^2$.
  Meta-learner picks meta-initialization ϕ_{t+1} using an OCO algorithm on $\ell_1, \ldots, \ell_t$.

unnecessary. In particular, record-level guarantees are likely to be sufficient whenever single records each pertain to different individuals. For example, for hospitals, what we care about is providing privacy to the individual patients and not aggregate hospital information. For cell-phones, if one can bound the number of texts that could contain the *same* sensitive information, then our setting can be straightforwardly extended to protect up to k records simultaneously.

Differential Privacy (DP). A de-facto standard privacy approach in machine learning has been to apply DP. Assuming a training set $D = \{z_i\}_{i=1}^m$, there are two common types of DP. In the first case, a randomized mechanism $\mathcal{M}$ is (ε, δ)-**globally differentially private** if for all measurable $\mathcal{S} \subseteq \text{Range}(\mathcal{M})$ and for all datasets D, D' that differ by at most one element:

$$\mathbb{P}[\mathcal{M}(D) \in \mathcal{S}] \leq e^{\varepsilon}\mathbb{P}[\mathcal{M}(D') \in \mathcal{S}] + \delta$$

If this holds for D, D' differing by at most k elements, then (ε, δ) *k-group DP* is achieved. On the other hand, a randomized mechanism $\mathcal{M}$ is (ε, δ)-**locally differentially private** if for any two possible training examples $z, z' \in \mathcal{X} \times \mathcal{Y}$ and measurable $\mathcal{S} \subseteq \mathcal{X} \times \mathcal{Y}$:

$$\mathbb{P}[\mathcal{M}(z) \in \mathcal{S}] \leq e^{\varepsilon}\mathbb{P}[\mathcal{M}(z') \in \mathcal{S}] + \delta$$

Global DP guarantees that it will be hard to infer the presence of a specific record in the training set by observing the output of $\mathcal{M}$. Meanwhile, local DP assumes a stronger threat model in which the *aggregator* who runs the algorithm also cannot be trusted.

DP in a Meta-Learning Setting. In meta-learning, both the meta-level sub-procedure, $\{\hat{\theta}_t\}_{t \in B} \to \phi$, and the within-task sub-procedure, $\{z_{t,i}\}_{i=1}^{m_t} \to \hat{\theta}_t$, can be considered individual queries and a DP algorithm can implement either to be DP. Further, for each query, the procedure may be altered to satisfy either local DP or global DP. Thus, there are four fundamental options.

(1) *Global DP:* Releasing ϕ will at no point compromise information regarding any specific $\hat{\theta}_t$.
(2) *Local DP:* Additionally, each $\hat{\theta}_t$ is protected from being revealed to the meta-learner.
(3) *Task-Global DP:* Releasing $\hat{\theta}_t$ will at no point compromise any specific $z_{t,i}$.
(4) *Task-Local DP:* Additionally, each $z_{t,i}$ is protected from being revealed to task-owner.

By invariance to post-processing, the guarantees for (3) and (4) also automatically apply to the release of any future iteration of ϕ, thus protecting against other task-owners as well. Meanwhile, though mechanisms (1) and (2) by definition protect individual $\hat{\theta}_t$, they actually mask the entire presence or absence of any task, thus satisfying a task-level threat model. Using this terminology we can categorize the previous works for DP in federated settings as we do in Table 1 in the supplement.

3 Differentially Private Parameter-Transfer

Our theoretical result is for the DP GBML method written in its online (regret) form in Algorithm 1. Observe that both within-task optimization and meta-optimization are done using some form of gradient descent. The key difference between this algorithm and traditional GBML is that since task-learners must send back privatized model updates, each now applies an DP SGD procedure to learn $\hat{\theta}_t$ when called. However, at meta-test time the task-learner will run a *non-private* SGD to obtain the parameter $\bar{\theta}_t$ used for inference, as this parameter does not need to be sent to the meta-learner. To obtain DP and learning guarantees, we use a variant of Algorithm 1 in which the (ε, δ)-DP algorithm is an noisy SGD procedure [4, Algorithm 1] and apply the corresponding stability analysis.

Following Baxter [5], we consider task-distribution samples $\mathcal{P}_1, \ldots, \mathcal{P}_T \sim \mathcal{Q}$ from some distribution $\mathcal{Q}$ and samples indexed by $i = 1, \ldots, m$ from those task-distributions to improve performance when

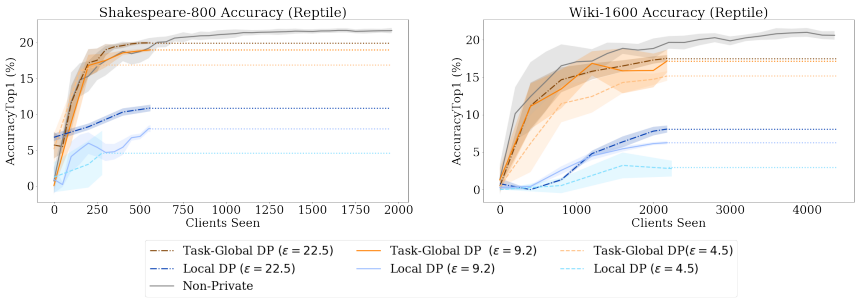

Figure 1: Performance of different versions of Reptile on a next-word-prediction task for two federated datasets. We report the test accuracy on unseen tasks and repeat each experiment 10 times. Solid lines correspond to means, colored bands indicate 1 standard deviation, and dotted lines are for comparing final accuracies (private algorithms can only be trained until privacy budget is met).

a new task $\mathcal{P}$ is sampled from $\mathcal{Q}$ and we draw m samples from it. Our goal will follow that of [21, 12] in seeking to obtain bounds on the transfer-risk – the distributional performance of a learned parameter on a new task from $\mathcal{Q}$ – that improve with task similarity. The specific notion that we consider is a simple one – that the risk-minimizing parameters of tasks sampled from the distribution $\mathcal{Q}$ are close together. This will be measured in-terms of the following quantity: $V^2 = \min_{\phi\in\Theta} \frac{1}{2}\mathbb{E}_{\mathcal{P}\sim\mathcal{Q}}\|\theta_{\mathcal{P}} - \phi\|_2^2$, for $\theta_P \in \arg\min_{\theta\in\Theta} \ell_{\mathcal{P}}(\theta)$ a risk-minimizer of task-distribution $\mathcal{P}$.

In the algorithm we study, each user t obtains a within-task parameter $\bar{\theta}_t$ by running (non-private) OGD on a sequence of losses $\ell_{t,1}, \ldots, \ell_{t,m}$ and averaging the iterates. The regret of this procedure, when averaged across the users, can be transformed using standard online-to-batch conversion Cesa-Bianchi et al. [10] into a bound on the expected excess transfer risk. Thus our goal is to bound this regret in terms of V; here we follow the ARUBA framework of Khodak et al. [22], to apply which we need a quadratic growth (α-QG) property on the within-task loss functions that essentially forces them to be non-degenerate, which we discuss in more detail in Appendix C. We are then able to state our main theoretical result, a proof of which is also in the supplement.

Theorem 3.1. *Suppose $\mathcal{Q}$ is a distribution over task-distributions $\mathcal{P}$ over Lipschitz, smooth, bounded and convex loss functions $\ell : \Theta \mapsto \mathbb{R}$ that satisfy a further weak regularity condition described in Bassily et al. [4] as well as α-QG. Suppose the distribution $\mathcal{P}_t$ of each task is sampled i.i.d. from $\mathcal{Q}$ and we run Algorithm 1 with the (ε, δ)-DP procedure of Bassily et al. [4, Algorithm 1] to obtain $\hat{\theta}_t$ as the average iterate for the meta-update step. Then if $\eta = V/\sqrt{m}$ for $V^2 = \min_{\phi\in\Theta} \frac{1}{2}\mathbb{E}_{\mathcal{P}\sim\mathcal{Q}}\|\theta_{\mathcal{P}} - \phi\|_2^2$ we have the following bound on the expected transfer risk when a new task $\mathcal{P}$ is sampled from $\mathcal{Q}$, m samples are drawn i.i.d. from $\mathcal{P}$, and we run OGD with learning rate η starting from $\bar{\phi} = \frac{1}{T}\sum_{t=1}^{T}\phi_t$ using the average $\bar{\theta}$ of the resulting iterates as the learned parameter:*

$$\mathbb{E}\underset{\mathcal{P}\sim\mathcal{Q}}{\mathbb{E}}\underset{\ell\sim\mathcal{P}}{\mathbb{E}}\ell(\bar{\theta}) \leq \underset{\mathcal{P}\sim\mathcal{Q}}{\mathbb{E}}\underset{\ell\sim\mathcal{P}}{\mathbb{E}}\ell(\theta^*) + \tilde{\mathcal{O}}\left(\frac{V}{\sqrt{m}} + \frac{1}{VT\sqrt{m}} + \frac{1}{V\alpha}\max\left(\frac{\sqrt{d\log\frac{1}{\delta}}}{\varepsilon m^{\frac{3}{2}}}, \frac{1}{m}\right)\right)$$

Here θ^* is any element of Θ and the outer expectation is over the data and the DP mechanism.

Theorem 3.1 shows that one can run a DP-algorithm as the within-task method in meta-learning and still obtain improvement due to task-similarity. Specifically, the standard term of $1/\sqrt{m}$ is multiplied by V, which is small if the tasks are related via the closeness of their risk minimizers. Thus we can use meta-learning to improve within-task performance relative to single-task learning. We also obtain fast convergence of $1/(VT\sqrt{m})$ in the number of tasks. This theorem also gives us a relatively straightforward extension for k-group-DP, as discussed in Appendix C.1.

4 Empirical Results

In this section, which we largely defer to the supplement, we present experiments that show that the relaxation to *task-global* DP allows for useful learning on federated language modelling tasks using Reptile to train a LSTM RNN. We demonstrate that on modified versions of Shakespeare [8] and Wikipedia-3029 [3], task-global DP training can achieve most of the performance of non-private training and dramatically more than *local* DP training.

References

[1] Martin Abadi, Andy Chu, Ian Goodfellow, Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *23rd ACM Conference on Computer and Communications Security (ACM CCS)*, pages 308–318, 2016.

[2] Naman Agarwal, Ananda Theertha Suresh, Felix Xinnan X Yu, Sanjiv Kumar, and Brendan McMahan. cpsgd: Communication-efficient and differentially-private distributed sgd. In *Advances in Neural Information Processing Systems 31*, pages 7564–7575. Curran Associates, Inc., 2018.

[3] Sanjeev Arora, Hrishikesh Khandeparkar, Mikhail Khodak, Nikunj Saunshi, and Orestis Plevrakis. A theoretical analysis of contrastive unsupervised representation learning. In *Proceedings of the 36th International Conference on Machine Learning*, 2019.

[4] Raef Bassily, Vitaly Feldman, Kunal Talwar, and Abhradeep Thakurta. Private stochastic convex optimization with optimal rates. URL `https://arxiv.org/abs/1908.09970`.

[5] Jonathan Baxter. A model of inductive bias learning. *Journal of Artificial Intelligence Research*, 12:149–198, 2000.

[6] Abhishek Bhowmick, John Duchi, Julien Freudiger, Gaurav Kapoor, and Ryan Rogers. Protection against reconstruction and its applications in private federated learning. arxiv, 2019.

[7] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H. Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for privacy preserving machine learning. Cryptology ePrint Archive, Report 2017/281, 2017. `https://eprint.iacr.org/2017/281`.

[8] Sebastian Caldas, Peter Wu, Tian Li, Jakub Konecný, H. Brendan McMahan, Virginia Smith, and Ameet Talwalkar. LEAF: A benchmark for federated settings. *CoRR*, abs/1812.01097, 2018. URL `http://arxiv.org/abs/1812.01097`.

[9] Nicholas Carlini, Chang Liu, Jernej Kos, Úlfar Erlingsson, and Dawn Song. The secret sharer: Measuring unintended neural network memorization & extracting secrets. *CoRR*, abs/1802.08232, 2018. URL `http://arxiv.org/abs/1802.08232`.

[10] Nicoló Cesa-Bianchi, Alex Conconi, and Claudio Gentile. On the generalization ability of on-line learning algorithms. *IEEE Transactions on Information Theory*, 50(9):2050–2057, 2004.

[11] Fei Chen, Zhenhua Dong, Zhenguo Li, and Xiuqiang He. Federated meta-learning for recommendation. *CoRR*, abs/1802.07876, 2018. URL `http://arxiv.org/abs/1802.07876`.

[12] Giulia Denevi, Carlo Ciliberto, Riccardo Grazzi, and Massimiliano Pontil. Learning-to-learn stochastic gradient descent with biased regularization. *CoRR*, abs/1903.10399, 2019. URL `http://arxiv.org/abs/1903.10399`.

[13] John Duchi, Martin Wainwright, and Michael Jordan. Minimax optimal procedures for locally private estimation. In *Journal of the American Statistical Association*.

[14] Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3&4):211–407, 2014. doi: 10.1561/0400000042.

[15] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, 2017.

[16] Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 1322–1333, 2015.

[17] Robin C. Geyer, Tassilo J. Klein, and Moin Nabi. Differentially private federated learning: A client level perspective, 2018. URL `https://openreview.net/forum?id=SkVRTj0cYQ`.

[18] Erin Grant, Chelsea Finn, Sergey Levine, Trevor Darrell, and Thomas L. Griffiths. Recasting gradient-based meta-learning as hierarchical bayes. *CoRR*, abs/1801.08930, 2018.

[19] Ghassen Jerfel, Erin Grant, Thomas L. Griffiths, and Katherine A. Heller. Online gradient-based mixtures for transfer modulation in meta-learning. *CoRR*, abs/1812.06080, 2018.

[20] Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the Polyak-Łojasiewicz condition. In *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, 2016.

[21] Mikhail Khodak, Maria-Florina Balcan, and Ameet Talwalkar. Provable guarantees for gradient-based meta-learning. In *Proceedings of the 36th International Conference on Machine Learning*, 2019.

[22] Mikhail Khodak, Maria-Florina Balcan, and Ameet Talwalkar. Adaptive gradient-based meta-learning methods. In *Advances in Neural Information Processing Systems 33*, 2019.

[23] Zhenguo Li, Fengwei Zhou, Fei Chen, and Hang Li. Meta-sgd: Learning to learn quickly for few shot learning. *CoRR*, abs/1707.09835, 2017.

[24] H Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. pages 1273–1282, 2017.

[25] H. Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. Learning differentially private language models without losing accuracy. *CoRR*, abs/1710.06963, 2017. URL `http://arxiv.org/abs/1710.06963`.

[26] Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *CoRR*, abs/1803.02999, 2018. URL `http://arxiv.org/abs/1803.02999`.

[27] Reza Shokri, Marco Stronati, and Vitaly Shmatikov. Membership inference attacks against machine learning models. *CoRR*, abs/1610.05820, 2016. URL `http://arxiv.org/abs/1610.05820`.

[28] Virginia Smith, Chao-Kai Chiang, Maziar Sanjabi, and Ameet Talwalkar. Federated multi-task learning. In *Advances in Neural Information Processing Systems 31*, 2017.

[29] Salvatore J. Stolfo, David W. Fan, Wenke Lee, Andreas L. Prodromidis, and Philip K. Chan. Credit card fraud detection using meta-learning: Issues and initial results 1. In *Working notes of AAAI Workshop on AI Approaches to Fraud Detection and Risk Management.*, 1997.

[30] Stacey Truex, Nathalie Baracaldo, Ali Anwar, Thomas Steinke, Heiko Ludwig, and Rui Zhang. A hybrid approach to privacy-preserving federated learning. *CoRR*, abs/1812.03224, 2018. URL `http://arxiv.org/abs/1812.03224`.

[31] Xi Sheryl Zhang, Fengyi Tang, Hiroko Dodge, Jiayu Zhou, and Fei Wang. Metapred: Meta-learning for clinical risk prediction with limited patient electronic health records, 2019. URL `https://arxiv.org/abs/1905.03218`.

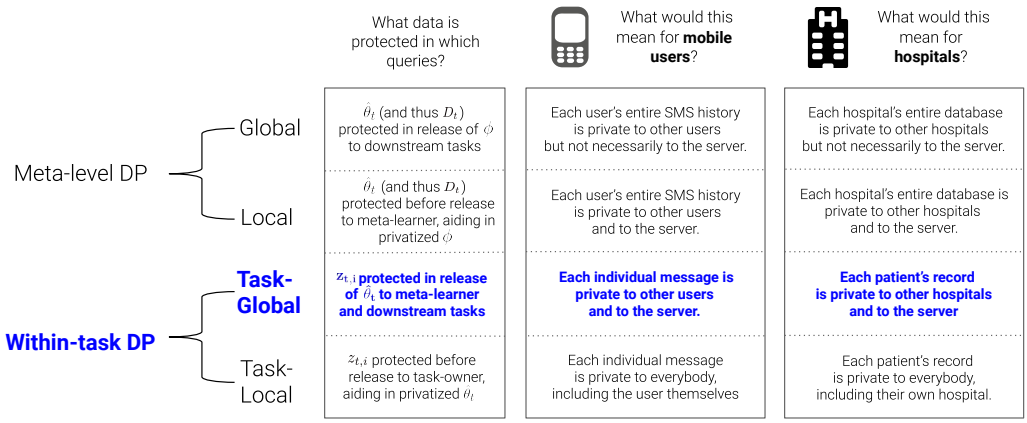

Figure 2: Summary of the privacy protections guaranteed by local and global DP at the different levels of the meta-learning problem (with our notion in blue). On the right, we show what each specification would mean in two practical federated scenarios: mobile users and hospital networks.

Table 1: Broad categorization of the DP settings considered by our work in meta-learning and notable past works in the federated setting. While these federated learning works do not assume a multi-task setting, we can still use the terms *global/local* and *task-global/task-local* to refer to the release of the global model and user-specific updates respectively. Overall, in contrast to past works, we observe that we are the first to formalize and consider privatizing the within-task algorithm.

Previous Work	Notion of DP	Privacy for ϕ	Privacy for θ_t
McMahan et al. [25]	Global	Task-level	-
Geyer et al. [17]	Global	Task-level	-
Bhowmick et al. [6]	Local, Global	Task-level	Task-level
Agarwal et al. [2]	Local + MPC	Task-level	Task-level
Truex et al. [30]	Task-Global + MPC	Record-level	Record-level
Our work	Task-Global	Record-level	Record-level

A Related Work

Privacy Attacks. Strongly motivating for our work are previous studies on the ability to infer specific training examples from a final machine learning model. In particular, [27] proposes a shadow training algorithm for membership inference attacks, being able to successfully discriminate whether any given example was present in a model's training set. Also, [9] shows that deep generative models can memorize unique patterns in training data, even if a *secret* were only seen once during training and regardless of over-fitting. This phenomenon is especially relevant to common parameter-transfer algorithms, given that they often do not repeatedly train on the same data. Though we do not directly test against such attacks and memorization, providing global DP by definition mitigates them.

Gradient-Based Meta-Learning Methods and Theory. The general learning problem of GBML is a broad approach for meta-learning that was popularized by MAML [15] and which has been extended in several directions [18, 19, 23, 26]. Most notably, Reptile [26] provides an SGD-based first-order simplification of the meta-update step in MAML. From a theoretical perspective, Khodak et al. [21] and Denevi et al. [12] provide finite-sample performance guarantees for GBML techniques in convex settings, focusing on Reptile-like and a strongly-regularized variant of Reptile, respectively. The learning analysis for our work extends the theory of the former work to focus on privacy.

DP Algorithms in Federated Learning Settings. Works most similar to ours focus on providing DP for federated learning. Specifically, Geyer et al. [17] and McMahan et al. [25] apply the gradient clipping and noising techniques of Abadi et al. [1] to achieve global DP federated learning algorithms for language modeling and image classification tasks, respectively. Their methods are shown to only suffer minor drops in accuracy compared to non-private training but they do not consider

protections to inferences made by the meta-learner. Alternatively, Bhowmick et al. [6] does achieve such protection by applying a theoretically rate-optimal local DP mechanism on the $\hat{\theta}_t$'s users send to the meta-learner. However, they sidestep hard minimax rates [13] by assuming limited adversaries have limited side-information and allowing for a large privacy budget. In this work, though we achieve a relaxation of the privacy of Bhowmick et al. [6], we do not restrict the adversary's power. Finally, Truex et al. [30] does assume a setting that coincides with our notion of task-global DP, but they focus primarily on the added benefits of applying MPC (see below) rather than studying the merits of task-global DP in comparison to other settings. Although these approaches all study privacy through the lens of learning a single global model, many of them, as well as our proposed GBML algorithm, are naturally amenable to a federated learning setting with personalization.

Secure Multiparty Computation (MPC). MPC is a cryptographic technique that allows parties to calculate a function of their inputs while also maintaining the privacy of each individual party's inputs [7]. In GBML, sets of model updates may come in a batch from multiple tasks, and hence MPC can securely aggregate the batch before it is seen by the meta-learner. Though MPC itself gives no DP guarantees, it prevents the meta-learner from directly accessing any one task's updates and can thus be combined with DP to increase privacy. Analogues of this approach have been studied in the federated setting, e.g. by Agarwal et al. [2], who apply SMC in the same difficult setting of Bhowmick et al. [6], and Truex et al. [30], who apply SMC similarly to a setting analogous to ours. On the other hand, MPC also comes with additional practical challenges such as peer-to-peer communication costs, drop outs, and vulnerability to collaborating participants. As such, combined with its applicability to multiple settings, including ours, we consider MPC to be an orthogonal direction.

B Local DP and Task-Global DP

Remark B.1. *If a GBML algorithm achieves (ε, δ)-local DP at the meta-level, it is also guaranteed to be (ε, δ)-DP at a task-global level.*

Proof. According to the definition of local DP, a mechanism $\mathcal{M}$ that achieves (ε, δ)-local DP for releasing ϕ must satisfy for any $\hat{\theta}_t, \hat{\theta}'_t \in \Theta$ and $\mathcal{S} \subseteq \Theta$:

$$\mathbb{P}[\mathcal{M}(\hat{\theta}_t) \in \mathcal{S}] \leq e^{\varepsilon}\mathbb{P}[\mathcal{M}(\hat{\theta}'_t) \in \mathcal{S}] + \delta$$

Here $\hat{\theta}_t$ can also be seen as a function, possibly stochastic, of D_t, or more formally, $\hat{\theta}_t = \mathcal{A}_\phi(D_t)$ where ϕ is an initialization and $\mathcal{A}_\phi : (\mathcal{X} \times \mathcal{Y})^m \to \Theta$. Thus, by also setting $\hat{\theta}'_t = \mathcal{A}_\phi(D'_t)$, we automatically get for any D_t, D'_t

$$\mathbb{P}[\mathcal{M}(\mathcal{A}_\phi(D_t)) \in \mathcal{S}] \leq e^{\varepsilon}\mathbb{P}[\mathcal{M}(\mathcal{A}_\phi(D'_t)) \in \mathcal{S}] + \delta$$

This holds by definition when $\mathcal{A}_\phi$ is deterministic since $\hat{\theta}_t$ and $\hat{\theta}'_t$ are single elements from Θ. When $\hat{\theta}_t$ and $\hat{\theta}'_t$ are stochastic, this bound also holds since it holds even in the worst case for any single pair of elements in Θ. Further, the bound holds no matter how many elements differ between D_t and D'_t, as long as $\mathcal{A}_\phi$ outputs something in Θ. Thus, if we treat $\mathcal{M}(\mathcal{A}_\phi(\cdot))$ as one mechanism, we get the given proposition.

□

C Proofs of Learning Guarantees

Throughout this section we assume all subsets are convex and in $\mathbb{R}^d$ unless explicitly stated. In the online learning setting we will use the shorthand ∇_t to denote the subgradient of $\ell_t : \Theta \mapsto \mathbb{R}$ evaluated at action $\theta_t \in \Theta$. For any $x_1, \ldots, x_T \in \mathbb{R}^d$ we will use $x_{1:t}$ to refer to the sum of the first t of them.

In this section we first prove (Theorem C.1) a general averaged-regret bound following the ARUBA framework of Khodak et al. [22]. We then combine an algorithmic stability based (ε, δ)-DP generalization bound for noisy SGD of Bassily et al. [4] with a quadratic growth assumption [20, 21] to

Algorithm 2: Online version of our (ε, δ)-meta-private parameter-transfer algorithm.

Meta-learner picks first meta-initialization $\phi_1 \in \Theta$.
for task $t \in [T]$ **do**
  Meta-learner sends meta-initialization ϕ_t to task t.
  Task-learner runs OGD starting from $\theta_{t,1} = \phi_t$ on losses $\{\ell_{t,i}\}_{i=1}^m$, suffering regret $\sum_{i=1}^m \ell_{t,i}(\theta_{t,i}) - \min_{\theta\in\Theta} \sum_{i=1}^m \ell_{t,i}(\theta)$.
  Task-learner t runs (ε, δ)-DP descent algorithm on losses $\{\ell_{t,i}\}_{i=1}^m$ to get $\hat{\theta}_t$.
  Task-learner sends $\hat{\theta}_t$ to meta-learner.
  Meta-learner constructs loss $\ell_t(\phi) = \frac{1}{2}\|\hat{\theta}_t - \phi_t\|_2^2$.
  Meta-learner picks meta-initialization ϕ_{t+1} using an OCO algorithm on $\ell_1, \ldots, \ell_t$.

show that such an algorithm returns a meta-update parameter $\hat{\theta}$ that is close θ^* and thus suffices to show a meaningful task-averaged-regret guarantee (Corollary C.1). We conclude by using this bound to derive a guarantee in the statistical LTL setting (Corollary C.2).

Setting C.1. *We assume all functions $\ell_{t,i} : \Theta \mapsto [0,1]$ are convex and G-Lipschitz for some $G \geq 1$ and that Θ has ℓ_2-diameter $D \geq 1$. We define the following quantities:*

- *convenience coefficients $\sigma = G\sqrt{m}$*
- *the sequence of update parameters $\{\hat{\theta}_t \in \Theta\}_{t\in[T]}$ with mean $\hat{\phi} = \frac{\hat{\theta}_{1:T}}{T}$*
- *a sequence of reference parameters $\{\theta'_t \in \Theta\}_{t\in[T]}$ with mean $\phi' = \frac{\theta'_{1:T}}{T}$*
- *a sequence $\{\theta^*_t \in \Theta\}_{t\in[T]}$ of optimal parameters in hindsight*
- *$\kappa \geq 1, \Delta^* \geq 0$ s.t. $\sigma \sum_{t=1}^T \mathbb{E}\,\|\theta^*_t - \phi_t\|_2^2 \leq \Delta^* + \kappa\sigma \sum_{t=1}^T \mathbb{E}\,\|\hat{\theta}_t - \phi_t\|_2^2$*
- *$\nu \geq 1, \Delta' \geq 0$ s.t. $\sigma \sum_{t=1}^T \mathbb{E}\,\|\hat{\theta}_t - \hat{\phi}\|_2^2 \leq \Delta' + \nu\sigma \sum_{t=1}^T \mathbb{E}\,\|\theta'_t - \phi'\|_2^2$*
- *positive task-similarity $V^2 = \frac{1}{2}\sum_{t=1}^T \mathbb{E}\,\|\theta'_t - \phi'\|_2^2$*
- *learning-rate $\eta = \frac{H}{G\sqrt{m}}$ for some $H > 0$*

Theorem C.1. *In Setting C.1 define the regret upper-bound $\hat{\mathbf{R}}_t = \frac{\|\theta^*_t - \phi_t\|_2^2}{2\eta} + \eta G^2 m$ and the averaged regret upper-bound $\hat{\bar{\mathbf{R}}} = \frac{1}{T}\sum_{t=1}^T \hat{\mathbf{R}}_t$. Then in Algorithm 2 if the meta-learner uses FTL or AOGD to pick the meta-initialization and the within-task descent algorithm has regret upper-bounded by $\hat{\mathbf{R}}_t$ we have the following bound:*

$$\mathbb{E}\,\hat{\bar{\mathbf{R}}} \leq \frac{\Delta^* + \kappa\Delta'}{HT} + \left(\frac{D^2\kappa}{2H}\frac{1+\log T}{T} + H + \frac{\kappa\nu V^2}{H}\right) G\sqrt{m}$$

Here the expectation is taken over the randomness of the DP mechanism.

Proof. We apply the standard FTRL regret of OGD, e.g. Theorem A.1 in Khodak et al. [21], and the logarithmic regret of FTL and AOGD, e.g. Theorem A.2 in Khodak et al. [21]:

$$\begin{aligned}
T\,\mathbb{E}\,\hat{\bar{\mathbf{R}}} &= \mathbb{E}\left(\sum_{t=1}^T \frac{\|\theta^*_t - \phi_t\|_2^2}{2\eta} + \eta G^2 m\right) \\
&= \frac{\Delta^*}{H} + \sigma\sum_{t=1}^T \mathbb{E}\left(\frac{\kappa}{2H}\|\hat{\theta}_t - \phi_t\|_2^2 + H\right) \\
&= \frac{\Delta^*}{H} + H\sigma T + \frac{\kappa\sigma}{2H}\sum_{t=1}^T \mathbb{E}\left(\|\hat{\theta}_t - \phi_t\|_2^2 - \|\hat{\theta}_t - \hat{\phi}\|_2^2 + \|\hat{\theta}_t - \hat{\phi}\|_2^2\right) \\
&\leq \frac{\Delta^*}{H} + H\sigma T + \frac{D^2\kappa\sigma}{2H}(1+\log T) + \frac{\kappa\Delta'}{H} + \frac{\kappa\nu\sigma}{2H}\sum_{t=1}^T \mathbb{E}\,\|\theta'_t - \phi'\|_2^2
\end{aligned}$$

□

Setting C.2. *In Setting C.1, assume loss functions $\ell_{t,1}, \dots, \ell_{t,m}$ are generated by picking some distribution $\mathcal{P}_t$ over valid losses and then sampling m of them i.i.d. Assume further that the expected loss of every such distribution satisfies α-quadratic-growth (α-QG): for some $\alpha > 0$, any $\theta \in \Theta$, and θ' the closest minimizer of $\mathbb{E}\,\ell$ to θ we have*

$$\frac{\alpha}{2}\|\theta - \theta'\|_2^2 \le \mathbb{E}(\ell(\theta) - \ell(\theta'))$$

Furthermore, assume that these losses are β-strongly-smooth:

$$\ell(\theta) \le \ell(\theta') + \langle \nabla \ell(\theta'), \theta - \theta' \rangle + \frac{\beta}{2}\|\theta - \theta'\|_2^2$$

Finally, assume that θ' is unique for every $\mathcal{P}_t$.

Lemma C.1. *Let $\ell_1, \dots, \ell_m : \Theta \mapsto [0,1]$ be a sequence of convex losses drawn i.i.d. from some distribution $\mathcal{D}$ with risk $\mathbb{E}\,\ell$ being α-QG and let $\theta^* \in \arg\min_{\theta\in\Theta} \sum_{i=1}^m \ell_i(\theta)$ be any of the optimal actions in hindsight. Then the closest minimum θ' of $\mathbb{E}\,\ell$ to θ^* satisfies*

$$\frac{1}{2}\,\mathbb{E}\,\|\theta^* - \theta'\|^2 \le \frac{2}{\alpha}\sqrt{\frac{1+\log m}{m}}$$

Proof. Taking expectations of the result of Lemma B.4 in Khodak et al. [21], we have for $\delta = \frac{2}{\sqrt{m}}$ that

$$\frac{\alpha}{2}\,\mathbb{E}\,\|\theta^* - \theta'\|_2^2 \le \sqrt{\frac{8}{m}\log\frac{2}{\delta}} + \delta \le \sqrt{\frac{4}{m}(1+\log m)}$$

□

Lemma C.2. *Let $\ell_1, \dots, \ell_m : \Theta \mapsto [0,1]$ be a sequence of β-strongly-smooth, G-Lipschitz convex losses drawn i.i.d. from some distribution $\mathcal{D}$ with risk $\mathbb{E}\,\ell$ being α-QG and let $\hat{\theta} \in \Theta$ be the average iterate of running Algorithm 1 of Bassily et al. [4] with the appropriate parameters for obtaining (ε, δ)-DP. If $\beta \le \frac{G}{D}\min\left(\frac{\sqrt{m}}{2}, \frac{\varepsilon m}{4\sqrt{d\log\frac{1}{\delta}}}\right)$ then the closest minimum θ' of $\mathbb{E}\,\ell$ to $\hat{\theta}$ satisfies*

$$\frac{1}{2}\,\mathbb{E}\,\|\hat{\theta} - \theta'\|_2^2 \le \mathbb{E}(\ell(\hat{\theta}) - \ell(\theta')) \le \frac{10GD}{\alpha}\max\left(\frac{\sqrt{d\log\frac{1}{\delta}}}{\varepsilon m}, \frac{1}{\sqrt{m}}\right)$$

Proof. The result follows by directly substituting Theorem 3.2 of Bassily et al. [4] into the definition of α-QG:

$$\frac{\alpha}{2}\,\mathbb{E}\,\|\hat{\theta} - \theta'\|_2^2 \le \mathbb{E}(\ell(\hat{\theta}) - \ell(\theta')) \le 10GD\max\left(\frac{\sqrt{d\log\frac{1}{\delta}}}{\varepsilon m}, \frac{1}{\sqrt{m}}\right)$$

□

Proposition C.1. *In Setting C.2 we have $\kappa = \nu = 3$ and*

$$\Delta^* = \frac{33G^2DT}{\alpha}\max\left(\sqrt{\frac{d}{\varepsilon^2 m}\log\frac{1}{\delta}}, \sqrt{1+\log m}\right) \quad \textit{and} \quad \Delta' = \frac{10G^2DT}{\alpha}\max\left(\sqrt{\frac{d}{\varepsilon^2 m}\log\frac{1}{\delta}}, 1\right)$$

Proof. We apply the triangle inequality, Jensen's inequality, and Lemmas C.1 and C.2 to get

$$\begin{aligned}
&\frac{\sigma}{2}\sum_{t=1}^{T}\mathbb{E}\,\|\theta_t^* - \phi_t\|_2^2 \\
&\quad\le \frac{3\sigma}{2}\sum_{t=1}^{T}\mathbb{E}\left(\|\theta_t^* - \theta_t'\|_2^2 + \|\theta_t' - \hat{\theta}_t\|_2^2 + \|\hat{\theta}_t - \phi_t\|_2^2\right) \\
&\quad\le 3\sigma\sum_{t=1}^{T}\left(\frac{6}{\alpha}\sqrt{\frac{1+\log m}{m}} + \frac{5GD}{\alpha}\max\left(\frac{\sqrt{d\log\frac{1}{\delta}}}{\varepsilon m}, \frac{1}{\sqrt{m}}\right) + \frac{1}{2}\mathbb{E}\,\|\hat{\theta}_t - \phi_t\|_2^2\right) \\
&\quad\le \frac{33G^2DT\sqrt{m}}{\alpha}\max\left(\frac{\sqrt{d\log\frac{1}{\delta}}}{\varepsilon m}, \sqrt{\frac{1+\log m}{m}}\right) + \frac{3\sigma}{2}\sum_{t=1}^{T}\mathbb{E}\,\|\hat{\theta}_t - \phi_t\|_2^2
\end{aligned}$$

We further have by the triangle inequality and Lemma C.2 that

$$\begin{aligned}
\frac{\sigma}{2}\sum_{t=1}^{T}\mathbb{E}\,\|\hat{\theta}_t - \hat{\phi}\|_2^2 &\le \frac{3\sigma}{2}\sum_{t=1}^{T}\mathbb{E}\left(\|\hat{\theta}_t - \theta_t'\|_2^2 + \|\theta_t' - \phi'\|_2^2 + \|\phi' - \hat{\phi}\|_2^2\right) \\
&\le 3\sigma\sum_{t=1}^{T}\mathbb{E}\,\|\hat{\theta}_t - \theta_t'\| + \frac{3\sigma}{2}\sum_{t=1}^{T}\mathbb{E}\,\|\theta_t' - \phi'\|_2^2 \\
&\le \frac{10G^2DT\sqrt{m}}{\alpha}\max\left(\frac{\sqrt{d\log\frac{1}{\delta}}}{\varepsilon m}, \frac{1}{\sqrt{m}}\right) + \frac{3\sigma}{2}\sum_{t=1}^{T}\mathbb{E}\,\|\theta_t' - \phi'\|_2^2
\end{aligned}$$

□

Corollary C.1. *In Setting C.2, if we run Algorithm 2 using OGD with learning rate H and Algorithm 1 of Bassily et al. [4] as the within-task (ε, δ)-DP method then for $H = V$ we have the following bound on the expected task-averaged regret:*

$$\mathbb{E}\,\bar{\mathbf{R}} \le \mathbb{E}\,\hat{\bar{\mathbf{R}}} = \frac{63G^2DT}{V\alpha}\max\left(\sqrt{\frac{d}{\varepsilon^2 m}\log\frac{1}{\delta}}, \sqrt{1+\log m}\right) + \frac{3GD^2\sqrt{m}}{V}\frac{1+\log T}{T} + 10GV\sqrt{m}$$

Proof. Substitute Proposition C.1 into Theorem C.1 and simplify. □

Corollary C.2. *In Setting C.2 and under the assumptions of Corollary C.1, if the distribution $\mathcal{P}_t$ of each task is sampled i.i.d. from some environment $\mathcal{Q}$ and then we have the following bound on the expected transfer risk when a new task $\mathcal{P}$ is sampled from $\mathcal{Q}$, m samples are drawn i.i.d. from $\mathcal{P}$, and we run OGD with $\eta = \frac{H}{G\sqrt{m}}$ starting from $\bar{\phi} = \frac{1}{T}\phi_{1:T}$ and use the average $\bar{\theta}$ of the resulting iterates as the learned parameter:*

$$\mathbb{E}\mathop{\mathbb{E}}_{\mathcal{P}\sim\mathcal{Q}}\mathop{\mathbb{E}}_{\ell\sim\mathcal{P}}\ell(\bar{\theta}) \le \mathop{\mathbb{E}}_{\mathcal{P}\sim\mathcal{Q}}\mathop{\mathbb{E}}_{\ell\sim\mathcal{P}}\ell(\theta^*) + \tilde{\mathcal{O}}\left(\frac{V}{\sqrt{m}} + \frac{D^2}{VT\sqrt{m}} + \frac{D}{V\alpha}\max\left(\frac{\sqrt{d\log\frac{1}{\delta}}}{\varepsilon m^{\frac{3}{2}}}, \frac{1}{m}\right)\right)$$

Here θ^* is any element of Θ and the outer expectation is taken over $\ell_{t,i} \sim \mathcal{P}_t \sim \mathcal{Q}$ and the randomness of the DP mechanism.

Proof. The result follows from two applications of the standard in-expectation online-to-batch argument, e.g. Proposition A.1 of Khodak et al. [21] followed by an application of Corollary C.1:

$$\begin{aligned}
\mathbb{E}\mathop{\mathbb{E}}_{\mathcal{P}\sim\mathcal{Q}}\mathop{\mathbb{E}}_{\ell\sim\mathcal{P}}\ell(\bar{\theta}) &\le \mathbb{E}\mathop{\mathbb{E}}_{\mathcal{P}\sim\mathcal{Q}}\left(\mathop{\mathbb{E}}_{\ell\sim\mathcal{P}}\ell(\theta^*)+\frac{\hat{\mathbf{R}}(\bar{\phi})}{m}\right)\\
&\le \mathop{\mathbb{E}}_{\mathcal{P}\sim\mathcal{Q}}\mathop{\mathbb{E}}_{\ell\sim\mathcal{P}}\ell(\theta^*)+\frac{\hat{\mathbf{R}}(\phi^*)}{m}+\frac{\mathbb{E}\,\hat{\tilde{\mathbf{R}}}}{Tm}\\
&\le \mathop{\mathbb{E}}_{\mathcal{P}\sim\mathcal{Q}}\mathop{\mathbb{E}}_{\ell\sim\mathcal{P}}\ell(\theta^*)+\tilde{\mathcal{O}}\left(\frac{V}{\sqrt{m}}+\frac{D^2}{VT\sqrt{m}}+\frac{D}{V\alpha}\max\left(\frac{\sqrt{d\log\frac{1}{\delta}}}{\varepsilon m^{\frac{3}{2}}},\frac{1}{m}\right)\right)
\end{aligned}$$

□

C.1 Extension to Group-DP

Since any privacy mechanism that provides (ε,δ)-DP also provides $(k\varepsilon, ke^{(k-1)\epsilon}\delta)$-DP guarantees for groups of size k [14], we immediately have largely the same learning rates except replacing.

Corollary C.3. *Under the same assumptions and setting as Theorem 3.1, achieving (ε,δ)-group DP is possible with the same guarantee except replacing $\sqrt{\frac{d}{\varepsilon^2}\log(\frac{1}{\delta})}$ with $\sqrt{k^2d+kd[\frac{1}{\epsilon^2}\log(\frac{k}{\delta})-1]}$.*

For constant k, this allows us to enjoy the stronger guarantee while maintaining largely the same learning rates. This is a useful result given that in some settings, it may be desired to simultaneously protect small groups of size $k << m_t$, such as protecting entire families for hospital records.

D Experiment Details

In this section, we present results that show it is possible to learn useful deep models in federated scenarios while still preserving *task-global* privacy. In particular, our focus is to evaluate the performance of models that have been optimzied with a *task-global* DP algorithm in comparison to models that are trained both non-privately and models that were trained with the previously more commonly studied *local* DP. To this end, we evaluate performance of a LSTM RNN for language modeling tasks and apply a practical variant of Algorithm 1 that considers both tasks and within-task examples in batches instead of serially. To obtain within-task privacy, we alter the within-task algorithm to be the DP-SGD algorithm form [1] and to obtain local privacy we use a modification of [25] where each task separately applies a Gaussian Mechanism on a single $\hat{\theta}_t$ before sending model updates to the meta-learner.

Datasets: We train a next word predictor for two federated datasets: (1) The Shakespeare dataset as preprocessed by [8], and (2) a dataset constructed from $3,000$ Wikipedia articles drawn from the Wiki-3029 dataset [3], where each article is used as a different task. For each dataset, we set a fixed number of tokens per task, discard tasks with fewer tokens than the specified, and discard samples from those tasks with more. We set the number of tokens per task to 800 for Shakespeare and to $1,600$ for Wikipedia, divide tokens into sequences of length 10, and we refer to these modified datasets as Shakespeare-800 and Wiki-1600. For Shakespeare-800, we leave 279 tasks for meta-training, 31 for meta-validation, and 35 for meta-testing. For Wikipedia-1600 we use $2,179$ tasks for meta-training, 243 for meta-validation, and 606 for meta-testing. For the meta-validation and meta-test tasks, 75% of the tokens are used for local training, and the remaining 25% for local testing.

Model Structure: Our model first maps each token to an embedding of dimension 200 before passing it through an LSTM of two layers of 200 units each. The LSTM emits an output embedding, which is scored against all items of the vocabulary via dot product followed by a softmax. We build the vocabulary from the tokens in the meta-training set and fix its length to $10,000$. We use a sequence length of 10 for the LSTM and, just as [25], we evaluate using `AccuracyTop1` (i.e., we only consider the predicted word to which the model assigned the highest probability) and consider all predictions of the unknown token as incorrect.

Meta Learning Algorithm. We study the performance of our method when applied to the batch version of Reptile [26] (which, in our setup, reduces to personalized Federated Averaging when the meta-learning rate is set to 1.0). We tune various configurations of task batch size for all training procedures and models and for the non-private baseline, we also tune for multiple visits per client since there is no privacy degradation to account for. Additionally, we implement an exponential decay on the meta learning rate.

Privacy Considerations. For the *task-global* DP models, we set $\delta = 10^{-3} < \frac{1}{m^{1.1}}$ on each task and we implement DP-SGD within-task using the tools provided by *TensorFlow Privacy*[1]. Although this algorithm differs from the one presented in Section 3, it lets us explore *task-global* privacy in a realistic setting. We use the the *RDP accountant* to track our privacy budgets. Finally, for the language modeling datasets, we make sure that all tasks are sampled without replacement with a fixed batch size until all are seen. This is necessary since multiple visits to a single client results in degradation of the privacy guarantee for that client. We instead aim to provide the same guarantee for each client. For local-DP, though this notion of DP is stronger, we explore the same privacy budgets so as to obtain guarantees that are of the same *confidence*. Here, we essentially run the DP-FedAvg algorithm from [25] with some key changes. First, to get local DP instead of global, we add Gaussian noise to each clipped set of model updates before returning them to the central server instead of after aggregation. Second, we again iterate through tasks without replacement.

Hyperparameters: We tune the hyperparameters on the set of meta-validation tasks. For all datasets and all versions of the meta-learning algorithm, we tune hyperparameters in a two step process. We first tune all the parameters that are not related to refinement: the meta learning rate, the local (within-task) meta-training learning rate, the maximum gradient norm, and the decay constant. Then, we use the configuration with the best accuracy pre-refinement and then tune the refinement parameters: the refine learning rate, refine batch size, and refine epochs.

All other hyperparameters are kept fixed for the sake of comparison: full batch steps were taken on within-task data, with the maximum number of microbatches used for the task-global DP model. The parameter search spaces are given in Tables 2, 3, 4.

Results. Figure 1 shows the performance of both the non-private and *task-global* private versions of Reptile [26] for the language modelling tasks across three different privacy budgets. As expected, neither private algorithm reaches the same accuracy of the non-private version of the algorithm. Nonetheless, the task-global version still comes within $78\%, 88\%,$ and 92% of the non-private accuracy for Shakespeare-800 and within $72\%, 82\%,$ and 83% for Wiki-1600. Meanwhile achieving local DP results in only about 50% and 39% of the non-private accuracy on both datasets for the *most* generous privacy budget. In practice, these differences can be toggled by further changing the privacy budget or trading off more training iterations for larger noise multipliers.

[1] https://github.com/tensorflow/privacy

Table 2: Hyperparameter Search Space for Non-Private Training

Hyperparameter	Shakespeare-800	Wiki-1600
Visits Per Task	$\{1, 2, 3, 4, 5, 6, 7, 8, 9\}$	$\{1, 2, 3\}$
Tasks Per Round	$\{5, 10\}$	$\{5, 10\}$
Within-Task Epochs	$\{1, 3, 5, 7, 9\}$	$\{1, 3, 5, 7, 9\}$
Meta LR	$\{1, \sqrt{2}, 2, 2\sqrt{2}, 4, 4\sqrt{2}, 8, 8\sqrt{2}\}$	$\{1, \sqrt{2}, 2, 2\sqrt{2}, 4, 4\sqrt{2}, 8, 8\sqrt{2}\}$
Meta Decay Rate	$\{0, 0.001, 0.005, 0.01, 0.025, 0.05, 0.1\}$	$\{0, 0.001, 0.005, 0.01, 0.025\}$
Within-Task LR	$\{\sqrt{2}, 2, 2\sqrt{2}, 4, 4\sqrt{2}, 8\}$	$\{1, \sqrt{2}, 2, 2\sqrt{2}, 4, 4\sqrt{2}, 8\}$
L_2 Clipping	$\{0.3, 0.5, 0.7, 0.8, 1.0\}$	$\{0.3, 0.5, 0.7, 0.8, 1.0\}$
Refine LR	$\{0.1, 0.15, 0.3, 0.5, 0.7, 0.8\}$	$\{0.1, 0.15, 0.3, 0.5, 0.7, 0.8\}$
Refine Mini-batch Size	$\{10, 20, 30, 60\}$	$\{10, 20, 30, 60, 120\}$
Refine Epochs	$\{1, 2, 3\}$	$\{1, 2, 3\}$

Table 3: Hyperparameter Search Space for Task-Global DP Training

Model	Shakespeare-800	Wiki-1600
Visits Per Task	$\{1, 2, 3\}$	$\{1, 2\}$
Tasks Per Round	$\{5, 10\}$	$\{5, 10\}$
Within-Task Epochs	1	1
Meta LR	$\{\sqrt{2}, 2, 2\sqrt{2}, 4, 4\sqrt{2}, 8, 8\sqrt{2}\}$	$\{1, \sqrt{2}, 2, 2\sqrt{2}, 4, 4\sqrt{2}, 8\}$
Meta Decay Rate	$\{0, 0.001, 0.005, 0.01, 0.025, 0.05, 0.1\}$	$\{0.0, 0.001, 0.005, 0.01, 0.025, 0.05\}$
Within-Task LR	$\{1, \sqrt{2}, 2\sqrt{2}, 4, 4\sqrt{2}, 8\}$	$\{1, \sqrt{2}, 2\sqrt{2}, 4, 4\sqrt{2}, 8\}$
L_2 Clipping	$\{0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}$	$\{0.3, 0.4, 0.5, 0.7, 0.8, 0.9, 1.0\}$
Refine LR	$\{0.1, 0.15, 0.3, 0.5, 0.7, 0.8\}$	$\{0.1, 0.15, 0.3, 0.5, 0.7, 0.8\}$
Refine Mini-batch Size	$\{10, 20, 30, 60\}$	$\{10, 20, 30, 60, 120\}$
Refine Epochs	$\{1, 2, 3\}$	$\{1, , 3\}$

Table 4: Hyperparameter Search Space for Local-DP Training

Model	Shakespeare-800	Wiki-1600
Visits Per Task	$\{1, 2, 3\}$	$\{1, 2\}$
Tasks Per Round	$\{5, 10, 20\}$	$\{10, 20, 40, 80\}$
Within-Task Epochs	$\{1, 2, 3\}$	$\{1, 2, 3\}$
Meta LR	$\{\sqrt{2}, 2, 2\sqrt{2}, 4, 4\sqrt{2}, 8, 8\sqrt{2}\}$	$\{2\sqrt{2}, 4, 4\sqrt{2}, 8, 8\sqrt{2}\}$
Meta Decay Rate	$\{0, 0.005, 0.01, 0.025, 0.05, 0.1\}$	$\{0.0, 0.001, 0.005, 0.01, 0.025, 0.05\}$
Within-Task LR	$\{1, \sqrt{2}, 2\sqrt{2}, 4, 4\sqrt{2}, 8\}$	$\{1, \sqrt{2}, 2\sqrt{2}, 4, 4\sqrt{2}, 8\}$
L_2 Clipping	$\{0.005, 0.01, 0.025, 0.05, 0.1, 0.25, 0.5\}$	$\{0.005, 0.01, 0.025, 0.05, 0.1, 0.25\}$
Refine LR	$\{0.1, 0.15, 0.3, 0.5, 0.7, 0.8\}$	$\{0.1, 0.15, 0.3, 0.5, 0.7, 0.8\}$
Refine Mini-batch Size	$\{10, 20, 30, 60\}$	$\{10, 20, 30, 60, 120\}$
Refine Epochs	$\{1, 2, 3\}$	$\{1, 2, 3\}$