Gone but Not Forgotten: Improved Benchmarks for Machine Unlearning

Keltin Grimes, Collin Abidi, Cole Frank, and Shannon Gallagher

Carnegie Mellon University
Software Engineering Institute
Pittsburgh, Pennsylvania 15213
{*kgrimes, cabidi, cfrank, skgallagher*}@*sei.cmu.edu*

Abstract—**Machine learning models are vulnerable to adversarial attacks, including attacks that leak information about the model's training data. There has recently been an increase in interest about how to best address privacy concerns, especially in the presence of data-removal requests. Machine unlearning algorithms aim to efficiently update trained models to comply with data deletion requests while maintaining performance and without having to resort to retraining the model from scratch, a costly endeavor. Several algorithms in the machine unlearning literature demonstrate some level of privacy gains, but they are often evaluated only on rudimentary membership inference attacks, which do not represent realistic threats. In this paper we describe and propose alternative evaluation methods for three key shortcomings in the current evaluation of unlearning algorithms. We show the utility of our alternative evaluations via a series of experiments of state-of-the-art unlearning algorithms on different computer vision datasets, presenting a more detailed picture of the state of the field.**

1. Introduction

While incorporating new data into established machine learning models via fine-tuning is a well-studied problem, the inverse problem of removing data from those models has received less attention until recently. The field of *machine unlearning* [1] [2] [3] has emerged over the last several years in response to legal frameworks, such as the EU's GDPR [4], that give citizens the "right to be forgotten." To comply with such legal frameworks, companies may be required not only to remove user data from data structures, but also from machine learning models trained on that data. Machine unlearning can be applied to many data domains including images, videos, audio, and text – in this paper we focus on computer vision (CV) in response to recent research efforts to comply with legal requirements for user privacy.

The naïve approach to unlearning would be to erase the required data from a dataset and retrain a model from scratch. The removed data will have no influence on the model parameters or its output (ignoring indirect influences such as hyper-parameters optimized prior to removal), so an attacker would have no way of extracting information about the removed data. The major downside of this approach - and the main issue that the machine unlearning literature

seeks to address - is the computational burden and hence the immense cost in terms of processes and technology. Retraining a model from scratch on unlearning requests that arrive in a stream may simply not be feasible at scale, especially for large datasets and models with hundreds of millions of parameters that require thousands of epochs on tens or hundreds of GPUs to train. Ideal unlearning algorithms would provide the privacy guarantees and performance characteristics of a retrained-from-scratch model and require far fewer compute resources. As such, machine unlearning involves balancing *privacy*, *cost*, and *long-term performance*.

To realistically represent conditions under practical attacks when benchmarking machine unlearning algorithms, we argue that researchers and practitioners should consider the following three essential characteristics:

1) Emphasis of *worst-case* metrics over average-case metrics and the use of strong adversarial attacks to provide a high-quality upper-bound on privacy.
2) Consideration of model *update*-based attacks (e.g. *leakage*), as in [5], which may cause unlearning to provide *additional* information to attackers.
3) Analysis of unlearning algorithm performance over repeated applications of unlearning (i.e. *iterative unlearning*), especially in regards to degradation of test accuracy performance of the unlearned models.

In response to these gaps, we propose a framework that incorporates a suite of improved benchmarks for the testing and evaluation of machine unlearning algorithms. To test our framework and demonstrate the utility of the improved benchmarks we conduct a benchmarking study of a variety of state-of-the-art (SoTA) unlearning algorithms with the goal of presenting a holistic assessment of unlearning algorithms. Our contributions are intended to demonstrate the importance of using more comprehensive evaluations.

1.1. Unlearning Definition and Notation

We are primarily interested in *approximate unlearning* where the influence of the forget set is practically removed, as opposed to exact unlearning where the influence must completely and provably removed. We define unlearning as follows. We have a dataset $\mathcal{D}$ partitioned into forget,

retain, validation, and test sets $\mathcal{D}_f$, $\mathcal{D}_r$, $\mathcal{D}_{val}$, and $\mathcal{D}_{test}$, respectively. A model M is trained on the training subset $\mathcal{D}_{train} = \mathcal{D}_f \cup \mathcal{D}_r$, where $\mathcal{D}_f \cap \mathcal{D}_r = \emptyset$. An unlearning algorithm $\mathcal{U} : M \times (\mathcal{D}_r, \mathcal{D}_f) \rightarrow M'$ produces a new model M' which ideally has minimized the possibility of information leakage from $\mathcal{D}_f$, maintained model performance, and done so with minimal compute. Iterative unlearning requires an extended definition (omitted here for brevity), but intuitively, it is the repeated act of unlearning over time.

2. Evaluating Unlearning Algorithms

Prior works in unlearning [5], [6], [7], [8], [9], [10], [11], [12], [13] take inspiration from the Differential Privacy (DP) literature, using membership inference attacks (MIAs), which can determine whether a certain example was part of the training set, to demonstrate an empirical upper bound on privacy. Intuitively, such privacy auditing attacks demonstrate what is possible by an adversary, and thus high-confidence upper bounds are crucial for understanding vulnerabilities in a system.

2.1. Privacy as a Worst-Case Metric

We argue that worst-case measures of privacy are crucial for effective evaluations of unlearning algorithms. Users, in the absence of a strict guarantee of their own privacy, will care primarily about worst-case outcomes for themselves, rather than average- or even best-case metrics. When examining privacy through the lens of DP, as many in the unlearning field do, this becomes even more clear. Satisfying (ϵ, δ)-Differential Privacy requires a mathematical proof showing an algorithm satisfies a strict worst-case upper-bound on potential information leakage from $\mathcal{D}_{train}$ across all possible training examples. Therefore, strong unlearning evaluations will both use as effective MIAs as possible, and present the results of the attacks with worst-case metrics.

Through our literature review we have found that the most common MIA used to evaluate algorithms is a simple Logistic Regression classifier trained to predict whether or not a loss value comes from an example in $\mathcal{D}_f$ [6], [14], [15], [16], [17]. Furthermore, results are reported almost exclusively through either accuracy or recall - both of which are average case metrics. The study in [16] takes an important step forward in using stronger MIAs, adapting the online version of the SoTA Likelihood Ratio Attack (LiRA) from [18] to the unlearning setting. The 'online' version refers to the need to train new models for every membership query. Notably, they continue the precedent set in [18] by showing Receiver Operating Characteristic (ROC) curves with log-scales to show performance at very low false-positive rates - better demonstrating worst-case outcomes. The effectiveness of LiRA as an MIA makes it a much more realistic estimate of privacy.

One downside of [16]'s online-LiRA implementation is its computational complexity. The attack entails training 256 'shadow' models (each on a random half of the training set) and then running an unlearning algorithm 10,000 times on random forget sets for *each* of the 256 shadow models. This amounts to over 2.5 million unlearning runs per algorithm. For any fine-tuning based algorithm, or any algorithm dealing with even moderately large models or data sets, this sort of evaluation is impractical.

To remedy this gap, **we adapt the *offline* version of LiRA for unlearning** in our benchmarking framework. 'Offline' means that no new shadow models need to be trained for new membership queries. This setup only requires a single unlearning run, making this attack much more feasible - especially in the iterative unlearning setting (Section 2.3). While it is not as strong an attack as online-LiRA, and therefore worse for privacy estimation, it is much stronger than the basic Logistic Regression MIA, so we recommend its use in cases where the online-LiRA attack is computationally infeasible.

While MIAs are often just used to rank unlearning algorithms, some approaches have been made to more directly estimate DP privacy parameters (ϵ, δ). The 2023 NeurIPS Machine Unlearning Competition [19] based their metric on group-level DP, a stricter formulation of DP which considers datasets differing by up to k examples, which makes sense for unlearning as requests are likely to be batched. In fact, the competition metric used estimates of worst-case MIA performance to estimate ϵ for *each* example in $\mathcal{D}_f$, producing an entire distribution of privacy levels. While [19] averaged the per-example estimates of ϵ into a single score, **we re-implement the per-example ϵ estimation to preserve the full distribution of privacy levels**, which we find to be a valuable point of comparison.

2.2. Update Leakage

While intuitively one may think that unlearning can only result in positive outcomes with respect to the privacy of forgotten data, [5] demonstrate that, in regards to an attacker who already has the outputs of the base model, the act of unlearning may result in *worse* privacy than not unlearning at all. The idea of model update-leakage - where an adversary uses the difference in behaviors of two models as an attack vector - has been previously studied in iterative learning scenarios [12], [20], and [5] demonstrate that the issue is still relevant in the unlearning setting. [5] find that update-leakages are often stronger than standard attacks for both fully retrained and unlearned models, although the effect is weaker for the latter. In this regard unlearning may actually be preferential to retraining in terms of privacy, as it could find an optimal middle ground between the two types of attacks. We therefore evaluate unlearning algorithms on update-leakage attacks to:

1) Demonstrate an additional benefit of unlearning over retraining from scratch by showing less susceptibility to update-leakage attacks.
2) Ensure a 'do no harm' criteria is satisfied by showing an update-leakage attack does no worse than an attack on the base model.

To the best of our knowledge (through review of citations of [5]), the only unlearning algorithm benchmarked on an

update-leakage attack, besides SISA in [5], is GraphEraser [21], an unlearning algorithm for graph data. **We implement and run evaluations on the update-leak attack from [5]**, which works very similarly to online-LiRA, except uses outputs from both the base and the unlearned models.

2.3. Iterative Unlearning

The final piece of a comprehensive unlearning evaluation is studying how unlearning affects a model after repeated applications of an algorithm. If unlearning can truly be a replacement for full retraining, and deployers of ML models fully comply with data removal requests and destroy the base model trained on $\mathcal{D}_f$, then unlearning must be *iteratively* applied to unlearned models. Few papers consider this set-up, and those that do are usually special cases where unlearning can be performed in closed form or with convex optimization [10] [11] [3]. Surprisingly, we were able to locate only one unlearning paper that evaluates iterative test accuracy [22]. We have not encountered an iterative test accuracy evaluation in the vision domain.

In the iterative setting, it is required at each iteration to both ensure effective forgetting and, crucially, maintain model performance, as performance degradation tends to accumulate over time. Single forget set evaluations simply do not capture this important dimension of real-world performance. It should be noted that auditing privacy over many iterations may be prohibitively expensive, especially for attacks like online-LiRA that require hundreds of unlearning runs per iteration - so in this setting, our offline-LiRA implementation might be preferred. In either case, test-set performance can be computed directly after each unlearning iteration, so we advocate for its inclusion in any unlearning evaluation. **We implement an iterative unlearning pipeline** that handles the required dataset splitting and model management to allow for extending existing unlearning evaluations to the iterative setting – we focus primarily on test-set accuracy.

2.3.1. Preliminary Results. We conduct a preliminary evaluation of various unlearning algorithms on the CIFAR10 dataset [23] with a ResNet18 architecture [24], which was chosen due to its small computational footprint. We evaluate the following baselines and unlearning algorithms: Identity (baseline), Retrain (baseline), Finetune (baseline, finetune on $\mathcal{D}_r$), RandLabel [8], BadTeach [14], SCRUB+R [16], SSD [6], and SSD+FT (SSD followed by Finetune).

For each unlearning iteration, $\mathcal{D}_f$ is constructed by sampling 1% of $\mathcal{D}_{train}$, conditioned on samples that have not already been forgotten in prior iterations. The sequence of forget sets is identical across all experiments. The Identity algorithm is a control where no unlearning is applied. The base model used to test each unlearning algorithm is a ResNet18 model trained for 30 epochs. The hyperparameters for each unlearning algorithm were discovered by running 100 trials with the Optuna framework [25], optimized for minimizing $|0.5 - \text{MIA Accuracy}|$ (aiming for the MIA to do no better than random) and maximizing validation accuracy.
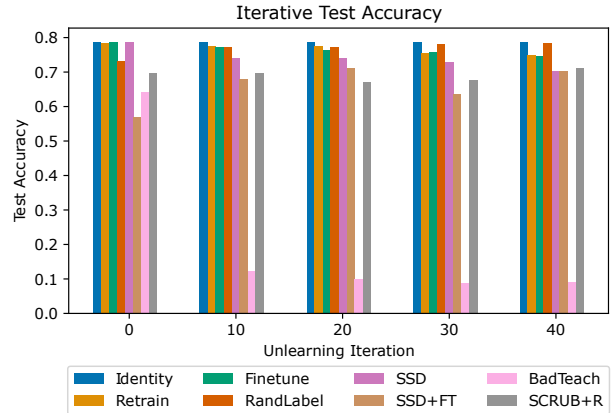

Figure 1. Iterative unlearning results for ResNet18 on CIFAR10.

In this iterative setting we find notable discrepancies in test accuracy amongst the various algorithms. As shown in Figure 1, BadTeach rapidly degrades model performance to random guessing, while other algorithms are able to maintain or in some cases increase accuracy over time. While these results only analyze test-set accuracy, and do not characterize either privacy or runtime, they demonstrate an important facet that we have observed to be considered in only one other unlearning evaluation [22].

3. Discussion

Overall, our initial tests show mixed results across unlearning algorithms, emphasizing the need for a holistic evaluation across all three major requirements (test accuracy, runtime, and privacy). For our privacy evaluations, we find that success in defending the basic Logistic Regression MIA, utilized in most unlearning evaluations, does *not* necessarily translate to success under stronger MIAs. In terms of update-leakage attacks, we find that only some of the tested algorithms perform better than retraining, but none have yet been found to leak extra information. As we saw in Section 2.3.1, there are considerable discrepancies in test accuracy in the iterative setting. We also find that hyperparameter tuning is a key aspect of algorithm performance, and different hyperparameters trade off performance and privacy differently. If optimizing hyperparameters for privacy with an expensive attack like online-LiRA the tuning process can be extremely slow, potentially presenting a significant barrier to real-world adoption.

4. Future Work

We will run evaluations on a variety of model architectures and datasets to provide a rigorous presentation of the state of the field. Our code base will be open sourced to provide an easy-to-use toolkit for designing and evaluating machine unlearning algorithms.

Acknowledgments

References

[1] H. Xu, T. Zhu, L. Zhang, W. Zhou, and P. S. Yu, "Machine unlearning: A survey," 2023.

[2] T. T. Nguyen, T. T. Huynh, P. L. Nguyen, A. W.-C. Liew, H. Yin, and Q. V. H. Nguyen, "A survey of machine unlearning," 2022.

[3] C. Guo, T. Goldstein, A. Hannun, and L. Van Der Maaten, "Certified data removal from machine learning models," *arXiv preprint arXiv:1911.03030*, 2019.

[4] "General data protection regulation," https://gdpr-info.eu/, accessed: 2023-12-07.

[5] M. Chen, Z. Zhang, T. Wang, M. Backes, M. Humbert, and Y. Zhang, "When machine unlearning jeopardizes privacy," in *Proceedings of the 2021 ACM SIGSAC conference on computer and communications security*, 2021, pp. 896–911.

[6] J. Foster, S. Schoepf, and A. Brintrup, "Fast machine unlearning without retraining through selective synaptic dampening," *arXiv preprint arXiv:2308.07707*, 2023.

[7] Y. Huang and C. L. Canonne, "Tight bounds for machine unlearning via differential privacy," 2023.

[8] L. Graves, V. Nagisetty, and V. Ganesh, "Amnesiac machine learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 13, 2021, pp. 11516–11524.

[9] A. Ginart, M. Y. Guan, G. Valiant, and J. Zou, "Making ai forget you: Data deletion in machine learning," 2019.

[10] V. Gupta, C. Jung, S. Neel, A. Roth, S. Sharifi-Malvajerdi, and C. Waites, "Adaptive machine unlearning," 2021.

[11] S. Neel, A. Roth, and S. Sharifi-Malvajerdi, "Descent-to-delete: Gradient-based methods for machine unlearning," 2020.

[12] S. Zanella-Béguelin, L. Wutschitz, S. Tople, V. Rühle, A. Paverd, O. Ohrimenko, B. Köpf, and M. Brockschmidt, "Analyzing information leakage of updates to natural language models," in *Proceedings of the 2020 ACM SIGSAC conference on computer and communications security*, 2020, pp. 363–375.

[13] J. Foster, K. Fogarty, S. Schoepf, C. Öztireli, and A. Brintrup, "Zero-shot machine unlearning at scale via lipschitz regularization," *arXiv preprint arXiv:2402.01401*, 2024.

[14] V. S. Chundawat, A. K. Tarun, M. Mandal, and M. Kankanhalli, "Can bad teaching induce forgetting? unlearning in deep networks using an incompetent teacher," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 6, 2023, pp. 7210–7217.

[15] A. Golatkar, A. Achille, and S. Soatto, "Forgetting outside the box: Scrubbing deep networks of information accessible from input-output observations," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIX 16*. Springer, 2020, pp. 383–398.

[16] M. Kurmanji, P. Triantafillou, and E. Triantafillou, "Towards unbounded machine unlearning," *arXiv preprint arXiv:2302.09880*, 2023.

[17] D. Choi and D. Na, "Towards machine unlearning benchmarks: Forgetting the personal identities in facial recognition systems," *arXiv preprint arXiv:2311.02240*, 2023.

[18] N. Carlini, S. Chien, M. Nasr, S. Song, A. Terzis, and F. Tramer, "Membership inference attacks from first principles," in *2022 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2022, pp. 1897–1914.

[19] E. Triantafillou, F. Pedregosa, J. Hayes, P. Kairouz, I. Guyon, M. Kurmanji, G. K. Dziugaite, P. Triantafillou, K. Zhao, L. S. Hosoya, J. C. S. Jacques Junior, V. Dumoulin, I. Mitliagkas, S. Escalera, J. Wan, S. Dane, M. Demkin, and W. Reade, "Neurips 2023 - machine unlearning," 2023. [Online]. Available: https://kaggle.com/competitions/neurips-2023-machine-unlearning

[20] A. Salem, A. Bhattacharya, M. Backes, M. Fritz, and Y. Zhang, "{Updates-Leak}: Data set inference and reconstruction attacks in online learning," in *29th USENIX security symposium (USENIX Security 20)*, 2020, pp. 1291–1308.

[21] M. Chen, Z. Zhang, T. Wang, M. Backes, M. Humbert, and Y. Zhang, "Graph unlearning," in *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, 2022, pp. 499–513.

[22] J. Chen and D. Yang, "Unlearn what you want to forget: Efficient unlearning for llms," 2023.

[23] A. Krizhevsky, G. Hinton *et al.*, "Learning multiple layers of features from tiny images," 2009.

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[25] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," 2019.