Defenses Against Membership Inference Attacks on Unlearned Data

Josep Domingo-Ferrer, Najeeb Jebreel, and David Sánchez

Universitat Rovira i Virgili,
Department of Computer Engineering and Mathematics,
CYBERCAT-Center for Cybersecurity Research of Catalonia,
Av. Països Catalans 26, 43007 Tarragona, Catalonia
{josep.domingo, najeeb.jebreel, david.sanchez}@urv.cat

Abstract. Machine unlearning is a discipline that seeks to make a machine learning model forget some of the information items it was trained on. This is a means to enforce the fundamental right-to-be-forgotten (RTBF), among others. However, unlearning may paradoxically increase the vulnerability of models to membership inference attacks (MIAs) on precisely the unlearned data, by creating distinguishable patterns between forgotten and non-forgotten data. This is a manifestation of the "Streisand effect", where attempts to remove or forget information make it more noticeable. In this paper, we investigate this critical challenge and propose defense mechanisms designed to seamlessly integrate with existing machine unlearning methods. These defenses aim to obscure the distinguishability of forgotten data, thus mitigating MIA risks without compromising model performance or computational efficiency. In addition, we define rigorous evaluation criteria to assess the effectiveness of such defenses.
Keywords: Privacy, Machine unlearning, Membership inference attacks, Right to be forgotten.

1 Introduction

Machine learning (ML) relies on vast amounts of data, often including sensitive personal information. This raises concerns about data privacy. In response, regulatory frameworks such as the European Union General Data Protection Regulation (GDPR) and the California Consumer Privacy Act (CCPA) have been established, enforcing the "right to be forgotten" (RTBF) as a fundamental privacy right [15,29].

Unlike in the RTBF for web content or search engines, simply removing the data to be forgotten is not enough in the case of machine learning. Trained models can memorize specific training examples and remain vulnerable to privacy attacks such as membership inference [35,42], model inversion [16], and training data extraction [8]. Complete retraining from scratch on the data to be retained (the original training data minus the data to be forgotten) after each deletion request is computationally expensive, especially for large-scale models and frequent requests.

Machine unlearning (MU) [6,5,32,3] offers a more efficient alternative solution, by eliminating the influence of specific data points without full retraining. Current approaches to MU can be broadly categorized into exact and approximate methods [3]. Exact unlearning aims to produce models that are statistically indistinguishable from those retrained from scratch without the forgotten data, while approximate methods provide weaker guarantees with better computational efficiency.

However, MU creates a paradox: deleting data can make them more detectable via membership inference attacks (MIAs), which aim to determine whether a specific data sample was used to train a

model [35]. This so-called "Streisand effect" in ML reveals a tension between unlearning and privacy, as forgetting leaves traces that expose the removed data [9,18]. In [9] it is shown that an attacker with black-box access to both the original model (pre-unlearning) and the unlearned model can compare their outputs on chosen inputs. Significant differences may disclose that a specific input was unlearned, implying that it was part of the original training set. Furthermore, [18] show that many unlearning methods can cause confidence scores of unlearned samples to exhibit distributions distinctly different from their distributions when they are unseen during training, creating another distinguishability signal that MIAs can exploit [7]. Moreover, recent work by [12] highlights that adaptive deletion requests —those that leverage outputs by previous unlearned models— exacerbate these risks, as standard unlearning guarantees do not protect privacy in such scenarios.

Differential privacy (DP,[14])-based MU methods have been proposed to address this issue. [12] contend that applying DP on retained records is necessary to protect deleted records under adaptive forgetting requests. DP-based MU methods formalize unlearning success through statistical indistinguishability between unlearned models and models trained exclusively on the data to be retained [19,12,10]. Noise injection and customized optimization techniques are used to achieve this indistinguishability. However, these techniques face three key limitations: (1) general degradation of model performance due to systematic noise addition, (2) limited applicability to non-convex models like deep neural networks (DNNS), and (3) impractical privacy/utility bounds for real-world deployment.

Contribution and plan of this article. We analyze the Streisand effect in MU, formalizing how unlearning facilitates MIAs against unlearned data. We introduce a suite of approaches to efficiently obscure whether a specific forget sample was used in training while retaining model utility. We also provide integration strategies that allow these defenses to complement existing unlearning methods without requiring the strict guarantees (and associated limitations) of DP.

The remainder of this paper is organized as follows. Section 2 provides background on MIAs, DP, and MU. Section 3 describes MIAs against unlearned data. In Section 4, we identify several criteria that must be met by a good defense against MIAs in MU. In Section 5, we sketch several potential defenses and evaluate them according to the previous criteria. Conclusions and future research directions are summarized in Section 6.

2 Background and preliminaries

2.1 Membership inference attacks

Membership inference attacks (MIAs) aim to determine whether a specific data point was included in the training dataset of an ML model [35,42,7]. Given a target ML model M trained on dataset D, and a target sample x, the adversary aims to determine whether $x \in D$ with significantly higher accuracy than random guessing.

MIAs exploit differences in the behavior of the ML model in training versus unseen data, such as higher confidence scores or lower loss for training samples [33,42,39]. Different MIAs can be defined according to the knowledge available to adversaries: black-box attacks only query model outputs, whereas white-box attacks assume access to model parameters or computations. Most practical MIAs assume black-box access and analyze loss values [42,39], confidence scores [7,2,43] or labels [11,27].

Several factors affect the vulnerability of a model to MIAs, including model complexity, training data size, and extent of overfitting [7,13]. Models with a higher capacity relative to the size of

the training data are generally more vulnerable. Defenses against MIAs typically involve data augmentation, regularization techniques, DP, or modifications to the training process to reduce these vulnerabilities.

2.2 Differential privacy

Differential privacy (DP [14]) provides a framework to quantify the privacy guarantees of algorithms in sensitive data. A randomized algorithm $\mathcal{M}$ satisfies (ε, δ)-DP if for all adjacent datasets D and D' differing in one data point, and for all output sets S we have $\Pr[\mathcal{M}(D) \in S] \leq e^{\varepsilon} \Pr[\mathcal{M}(D') \in S] + \delta$. Here, ε is the privacy budget that controls privacy strength, and δ is a small probability of failure. When $\delta = 0$, we have pure ε-DP; otherwise, approximate DP applies. Notable DP mechanisms for ML include DP-SGD [1] (gradient clipping and noise) and PATE [30] (teacher-student ensembles).

DP is the most commonly used defense against MIAs as it limits how much information an attacker can learn about whether specific data were used in training [42].

2.3 Machine unlearning

Machine unlearning (MU) refers to the process of removing the influence of specific data points from a trained ML model while preserving the overall functionality [6,5]. Unlearning approaches can be categorized into two main types: exact and approximate unlearning.

Exact unlearning. It means that the unlearned model does not contain traces of the data to be forgotten [6,5], making it virtually equivalent to a model re-trained from scratch without the data to be forgotten. Although exact unlearning provides the strongest guarantee, it often requires significant retraining or *ad-hoc*/complex model architectures designed specifically for efficient unlearning. A pioneer exact unlearning method is SISA [5], which partitions data into shards and slices, and saves checkpoints during training to enable more efficient re-training/unlearning.

Approximate unlearning. It offers a more cost-effective alternative, which is particularly beneficial for complex models and frequent unlearning requests. Most of the approximate unlearning methods modify the trained model without full retraining, achieving efficient but not guaranteed unlearning [23,21,40,25,41,44]. The validation of these methods is thus based on empirical evidence, such as their robustness to MIAs. For example, gradient ascent-based approaches attempt to revert the parameter updates caused by specific forget examples by maximizing their loss during unlearning [18,41]. Influence function-based unlearning quantifies the influence of data on models, estimating the changes needed when erasing data, and efficiently updating the model accordingly [45]. Unlearning based on knowledge distillation trains a student model by selectively replicating knowledge from the original teacher model, which tends to remove sensitive information while preserving utility [25]. [26] employ this approach to protect the data of electric vehicle users using dual-term loss functions.

Another class of approximate methods provides formal guarantees on the removal of the influence of certain data. The most common definition, proposed by [19], is inspired by DP:

$$\Pr[\mathcal{U}(\mathcal{A}(D), D, S) \in O] \leq e^{\varepsilon} \Pr[\mathcal{A}(D \setminus S) \in O] + \delta,$$

where $\mathcal{A}$ is the learning algorithm, $\mathcal{U}$ is the unlearning algorithm, D is the training dataset, and S is the set of samples to be unlearned. This definition ensures that the unlearned model is statistically indistinguishable from a model retrained from scratch.

Similar definitions include Rényi unlearning [10], which uses Rényi divergence to measure the distance between distributions, and total variation stability [38], which bounds the total variation distance between model distributions.

The relationship between DP and unlearning has been explored by several researchers. [34] showed that DP implies a form of unlearning guarantee, and that a model trained with (ε, δ)-DP guarantees can be used to unlearn up to $O(n/\sqrt{d})$ samples while maintaining good generalization, where n is the size of the dataset and d is the dimension parameter. [12] demonstrated that DP for the records to be retained is necessary to ensure the privacy of deleted records under adaptive forgetting requests. Therefore, DP-based methods can be employed to counter MIAs in unlearning. However, these methods have several limitations, as we pointed out in Section 1.

3 Membership inference attacks against unlearned data

MU, though intended to protect privacy, can inadvertently make unlearned samples more vulnerable to MIAs. This paradoxical situation is analogous to the Streisand effect [12], where efforts to remove data lead to its increased detectability relative to a scenario in which which no removal actions have been taken.

The core of the vulnerability lies in the discrepancy between the original and the unlearned models as well as in the unusual behavior of the unlearned model on forgotten data. In a typical setting, a model $M = \mathcal{A}(D)$ is trained on a dataset $D = \{x_1, x_2, \ldots, x_n\}$ and later modified using an unlearning algorithm $\mathcal{U}$ to produce the unlearned model $M' = \mathcal{U}(M, D, S)$, where $S \subseteq D$ is the set of samples to be removed. As shown in [9], the difference between the outputs of M and M' can be exploited: if a sample x belongs to S, the unlearned model tends to produce a significantly altered prediction compared to the original model. Conversely, if $x \notin S$, the outputs of both models remain largely similar. This behavior sets the ground for MIAs even when exact unlearning (by retraining on the retained data) is employed. [9] leverage outputs from both M and M' by concatenating or differencing their posterior distributions. Empirical studies have shown that such combined attacks can yield an area under the curve (AUC) of 0.89, a significant increase compared to the roughly 0.5 AUC observed for classical MIA methods [35,33] on well-generalized models. In scenarios where only predicted labels are available, label-only attacks have been proposed [28], which infer membership by analyzing shifts in decision boundaries between M and M'.

Furthermore, unlearning methods can induce unusual behavior on forgotten samples. As noted in [12,18], the confidence scores of the unlearned model on such samples may follow a distribution markedly different from that of a model that never encountered them.

Additional risks arise in more dynamic settings. When multiple unlearning requests are processed sequentially, each intermediate model version reveals extra information that can be aggregated by attackers. Similarly, when a group of samples is deleted simultaneously or in adaptive deletion scenarios —where deletion requests are informed by previous model outputs— the cumulative effect can significantly amplify disclosure risks [9,12,20].

To mitigate these vulnerabilities, several strategies have been proposed, such as restricting output information by publishing only predicted labels rather than full confidence scores, applying temperature scaling to soften softmax outputs, or incorporating DP into the unlearning process [19]. In addition, privacy-aware unlearning methods that take advantage of membership inference as a guide to remove privacy-sensitive information from both weights and activation values have been introduced [37]. [12] demonstrated that DP for the records to be retained is necessary to avoid disclosing deleted records under adaptive forgetting requests.

4 Evaluation criteria for defenses against MIAs on unlearned data

Effective defenses against MIAs on unlearned data must balance several competing objectives. We propose the following evaluation criteria for such defenses:

1. *Protection of privacy.* The fundamental goal of any defense is to prevent information leakage about training data. Privacy can be measured through statistical metrics (*e.g.*, empirical AUC or MIA precision) or formal guarantees (meaningful DP bounds). A robust defense should ensure that any MIA performs no better than random guessing —for instance, achieving an AUC near 50%. Additionally, it should maintain a negligible number of true positives at extremely low false positive rates (FPR, e.g., below 0.1%), thereby reducing an attacker's certainty (precision) in positive inferences. Without such strong privacy protections, the RTBF is compromised, as attackers could still reliably identify training data members.
2. *Preservation of the utility of the model.* Defenses must maintain predictive performance and generalization capability. Excessive utility degradation renders unlearning methods impractical for real-world deployment. If privacy comes at the cost of making models unusable, organizations will be reluctant to implement unlearning, which will ultimately harm privacy rights.
3. *Computational efficiency.* Defense mechanisms should add minimal overhead to the already computationally intensive unlearning process. [12] demonstrated that inefficient defenses can increase computational costs by orders of magnitude, making them impractical for large models or sequential unlearning requests.
4. *Robustness against adaptation.* Defenses must remain effective against attackers who continually refine their strategies based on observed model changes. Adaptive attackers improve their methods over time by analyzing defense patterns. A defense that works only against static attacks will eventually fail. [20] showed that adaptive adversaries can exploit sequential model releases to extract significantly more information than static adversaries.
5. *Scalability.* Defense mechanisms should work efficiently with large-scale models and integrate easily into existing ML pipelines. Modern ML systems use large-scale models that contain billions of parameters. Defenses that work only for small models or require specialized infrastructure will have limited practical impact.

Although tradeoffs among them are inevitable in practice, these criteria enable systematic comparison of defense approaches, highlighting their relative strengths and limitations.

5 Proposed defenses against MIAs on unlearned data

We propose multiple defense approaches to mitigate MIAs against unlearned data. Each mechanism addresses the evaluation criteria of Section 4 in different ways.

5.1 Calibrated perturbation (CP)

This defense introduces carefully calibrated noise into the model output during inference, derived from the observed differences in predictions on forgotten samples before and after unlearning. By applying these perturbations to random samples of unseen test data with a controlled probability, CP obscures the distinguishability of forgotten samples while preserving model utility. CP operates as follows:

1. *Perturbation extraction.* After the unlearning process, compute the prediction difference for each forgotten sample $x_f \in S$ as $\Delta_f = M(x_f) - M'(x_f)$, where $M(x_f)$ and $M'(x_f)$ represent the output probabilities (or logits) of the original and unlearned models, respectively. If unlearning has not yet occurred, the defender can simulate the unlearning process on a random subset of the training data.
2. *Perturbation distribution modeling.* Aggregate the differences $\{\Delta_f\}$ to estimate their statistical properties (*e.g.*, mean μ and covariance Σ). Define a perturbation distribution $\mathcal{P}$, such as a multivariate Gaussian $\mathcal{N}(\mu, \Sigma)$ or a uniform distribution over the observed Δ_f.
3. *Controlled perturbation application.* During inference, for each input x, with probability p, sample a perturbation $\delta \sim \mathcal{P}$ and adjust the output of the unlearned model as $\tilde{M}'(x) = M'(x) + \delta$. With probability $1 - p$, leave the output as $\tilde{M}'(x) = M'(x)$. Note that $M'(x)$ can represent either logits or the final confidence scores; when it is the latter, the defender must normalize $\tilde{M}'(x)$ so that the scores sum to 1.

In terms of our evaluation criteria, this approach offers several key advantages:

- Privacy protection: CP renders the outputs of forgotten samples statistically indistinguishable from those of non-forgotten ones, thereby reducing the precision of MIAs and providing plausible deniability to the individuals requesting forgetting.
- Utility preservation: The probability p and the scale of the disturbance can be tuned to limit accuracy degradation. For example, taking p proportional to the relative size of the forgotten set compared to the total training data and clipping δ to minimally affect the predictions can achieve a balance between privacy and utility.
- Computational efficiency: Sampling and applying perturbations adds negligible overhead during inference.
- Robustness against adaptation: CP is robust against adaptation when the perturbation distribution $\mathcal{P}$ is complex and tied to the unlearning process, and when the probability p is carefully chosen to limit the attacker's ability to infer or bypass the noise. However, its robustness depends on the defender's ability to keep $\mathcal{P}$ and p confidential and to dynamically adapt the defense if needed. Against highly sophisticated attackers with extensive query access, additional measures (*e.g.*, differential privacy or randomized response) may be required to complement CP.
- Scalability: CP is compatible with any model architecture and can be implemented as an efficient wrapper around the model API, making it suitable for large-scale deployment.

5.2 Nearest neighbor confidence calibration (NNCC)

The key insight behind NNCC is that the responses of the unlearned model or the difference before and after unlearning can reveal membership information through subtle patterns in the outputs [7]. By mixing direct model outputs with aggregated responses from similar samples in a held-out dataset, we can mask these patterns and reduce the distinguishability between forgotten and non-forgotten data.

Similarly to CP, this approach works as a black-box protection layer over the unlearned model M', and aims to obscure the statistical signatures of unlearning without requiring any modification of the model parameters. Instead of returning the confidence score of the forgotten sample $M'(x_f)$, we calibrate its confidence scores $M'(x_f)$ using a reference dataset, D_{unseen}, unseen by M during training. We assume the availability of D_{unseen} to the defender. Then, for each forgotten sample $x_f \in S$, we:

– Identify the k nearest neighbors $\mathcal{N}_k(x_f) = \{x_1, x_2, \ldots, x_k\} \subset D_{unseen}$ in the feature space using a distance metric $d(x_f, x_i)$ (*e.g.*, Euclidean or cosine).
– Compute the target confidence distribution as an aggregation of the model outputs on x_f and these neighbors:

$$p_{target}(x_f) = \lambda \cdot M'(x_f) + (1 - \lambda) \cdot \frac{1}{k} \sum_{i=1}^{k} M'(x_i).$$

– Return $p_{target}(x_f)$ instead of $M'(x_f)$.

Assuming that M' is not overfitted, the confidence scores of the k nearest neighbors are not expected to be significantly different from those of x_f before forgetting. This makes the difference in confidence before and after forgetting less noticeable to MIAs.

On the other hand, calibrating the prediction confidence of x_f with those of similar unseen data keeps the new confidence distribution similar to that of genuine unseen data. The parameter $\lambda \in [0, 1]$ helps balance the trade-off between preserving the original confidence scores and hiding the forgetting signature. A smaller λ provides stronger privacy protection but potentially at the cost of reduced utility, while a larger λ better preserves the unlearned model's original behavior.

The evaluation of NNCC against our criteria follows:

– Privacy protection: By calibrating with similar unseen data, NNCC reduces the distinguishability of forgotten samples, making MIAs less effective.
– Utility preservation: Since calibration uses semantically similar samples, model responses remain meaningful and preserve the general patterns learned from training data.
– Computational efficiency: NNCC requires only nearest-neighbor computations and weighted averaging, making it computationally efficient.
– Robustness against adaptation: NNCC mitigates adaptive attacks by calibrating outputs with semantically similar reference samples, effectively masking the patterns required by adaptive attacks on forget samples. However, its effectiveness depends on the quality and representativeness of the reference dataset D_{unseen}. If this dataset is not representative, the calibration may be less effective against adaptive attacks.
– Scalability: NNCC is model-agnostic and can be implemented as a wrapper around model APIs, making it suitable for various deployment scenarios.

5.3 NNCC with KL regularization

Unlike the *post-hoc* NNCC approach described in Section 5.2, KL-regularized NNCC (KL-NNCC) directly embeds the calibration mechanism into the unlearning process. KL-NNCC modifies the model parameters by augmenting the unlearning objective of any MU method with a Kullback-Leibler (KL) divergence regularization term that guides the model's behavior on forgotten samples toward that of similar unseen samples. The procedure consists of the following steps:

1. Neighbor-based calibration: For $x_f \in S$, compute $p_{\text{target}}(x_f) = \frac{1}{K} \sum_{i=1}^{K} M'(x_i)$.
2. Optimize unlearning as

$$\min_{\phi} \lambda \mathcal{L} + (1 - \lambda) \sum_{x_f \in S} D_{\text{KL}} \left(\hat{M}'(x_f) \parallel p_{\text{target}}(x_f) \right). \tag{1}$$

In Equation (1), ϕ are the unlearned model parameters and the KL divergence term ensures calibration fidelity of the unlearned model to the neighborhood distribution. In terms of the evaluation criteria:

- Privacy protection: By directly embedding calibration into model parameters, KL-NNCC provides stronger privacy protection against potential white-box MIAs.
- Utility preservation: Parameter λ in Equation (1) allows control over the trade-off between utility (the original loss term $\mathcal{L}$) and privacy (KL divergence term).
- Computational efficiency: Additional optimization is required during unlearning, making it more computationally intensive than *post-hoc* methods described above.
- Robustness against adaptation: As with standard NNCC, robustness depends on the representativeness of the reference dataset used to compute target distributions.
- Scalability: Requires modification to the unlearning process, potentially limiting scalability compared to *post-hoc* approaches.

5.4 Input pre-processing

If the model unlearns x_n, then it can keep a one-way hash image $F(x_n)$ together with the output probabilities and the loss that corresponded to x_n before unlearning. Then, for any input, the model first computes its hash. If it coincides with $F(x_n)$, the model returns the output probabilities and loss stored for $F(x_n)$, perhaps with some small random difference. The goal is that the behavior of M and M' for a forgotten item x_n is sufficiently similar to prevent an attacker from inferring that x_n has been forgotten.

From the point of view of our evaluation criteria:

- Privacy protection: A consistent behavior is maintained for forgotten samples, making it difficult for attackers to detect unlearning. Note that remembering $F(x_n)$ is not remembering x_n, which can still be considered unlearned. However, if the model is open-source, the code to check the hash and return the stored probabilities and loss could certainly be removed by an attacker.
- Utility preservation: High utility is preserved as original model output is maintained for forgotten samples.
- Computational efficiency: Only one hash computation and lookup are required, making it computationally efficient.
- Robustness against adaptation: A form of indistinguishability is achieved between the original and unlearned models for forgotten samples. However, sophisticated attackers who can observe multiple outputs might detect subtle patterns in the "small random differences" added to stored outputs, potentially allowing them to identify forgotten samples through statistical analysis.
- Scalability: This defense is easily implementable across various model architectures and deployment scenarios.

5.5 Randomized training/fine-tuning

MIAs against unlearned data rely on comparing the behavior of the models before and after unlearning. This comparison can be thwarted by incorporating some randomization in the unlearning process, so that the behavior of M and M' may be different for retained items. In this case, when the attacker detects a different behavior of M and M' for a certain item, they cannot decide whether the item is forgotten or retained. At least two options arise:

– DP-certified unlearning fully adheres to the randomization principle.
– A less random and, therefore, more utility-preserving alternative is to resort to ML anti-overfitting techniques [4], which reduce the chance that ML models memorize sample-specific details that may result in distinct behaviors for training data (*members*) versus unseen data (*non-members*). Specific methods are weight regularization [24], weight dropout [36], data augmentation [46], learning rate tuning [22], and early stopping [31].

We now evaluate randomized training/fine-tuning against our criteria:

– Privacy protection: Strong privacy guarantees require a lot of noise injection in the unlearning stage. If DP is used, sequential unlearning degrades guarantees, as more items are unlearned, due to sequential composition. For sequential unlearning, fine-tuning on the remaining data provides a better utility-privacy trade-off. The theoretical guarantees of this approach often do not translate well to practical non-convex models like deep neural networks. If anti-overfitting is used, there are no formal privacy guarantees.
– Utility preservation: There is a trade-off with privacy (more noise means less utility).
– Computational efficiency: Additional training steps are required due to slower convergence.
– Robustness against adaptation: Randomization creates a form of statistical independence that helps defend against adaptive attackers, but requires careful calibration.
– Scalability: It may be less effective for large models, where fine-tuning is more resource-intensive.

5.6 Computational complexity of exhaustive MIAs

Finally, an intrinsic defense is the computational complexity of MIAs. Using them to discover which data were unlearned requires an exhaustive search. We have two situations:

– If the items are records whose attributes take values in relatively restricted ranges, then mounting an MIA for each possible combination of values may be feasible.
– If the items are relatively long unstructured text or images, mounting an MIA for each potential text or image instantiation is hardly feasible.

We can give the following information-theoretic argument to illustrate the complexity of such an exhaustive search. Assume that the information content of the item X (record, unstructured text, image, etc.) that has been potentially unlearned is $H(X)$ bits, where H is Shannon's entropy. Then discovering X by exhaustive search is equivalent to discovering a random cryptographic key of $H(X)$ bits. If $H(X)$ is, say, 64 or more bits, this is known to be computationally infeasible.

The evaluation of the computational complexity defense against our criteria follows:

– Privacy protection: Protection level varies by data type, with high-entropy data (like images) receiving strong protection while structured data with limited possible values receiving less.
– Utility preservation: Utility preservation is perfect as no model modification is required.
– Computational efficiency: No additional computations are required for implementation.
– Robustness against adaptation: Robustness is high as it is based on fundamental information-theoretic constraints rather than specific defense patterns. However, sophisticated attackers might develop more efficient search strategies or heuristics that reduce the effective search space, particularly if they can make adaptive queries to guide their search.
– Scalability: It is high as no implementation is needed beyond standard unlearning protocols.

6 Conclusions and future research

The paradoxical relationship between MU and privacy presents a significant challenge for privacy-preserving machine learning. Although unlearning aims to enhance privacy by removing data influence, it can inadvertently create even more serious privacy vulnerabilities through the very process of data removal. In response, we have proposed several defense mechanisms that can be incorporated into existing unlearning methods to make them more resilient to MIA.

Future research directions include developing unlearning methods with formal privacy guarantees, creating more robust defenses against unlearning-based attacks, and exploring the trade-offs between unlearning efficiency, model utility, and privacy protection [17].

Acknowledgments

This work was funded by the Government of Catalonia (ICREA Acadèmia Prizes to J. Domingo-Ferrer and to D. Sánchez), MCIN/AEI/ 10.13039/501100011033 and "ERDF A way of making Europe" under grants PID2021-123637NB-I00 "CURLING" and PRE2019-089210, and INCIBE and European Union NextGenerationEU/PRTR (project "HERMES" and INCIBE-URV Cybersecurity Chair).

References

1. 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*, pages 308–318, 2016.
2. Martin Bertran, Shuai Tang, Aaron Roth, Michael Kearns, Jamie H Morgenstern, and Steven Z Wu. Scalable membership inference attacks via quantile regression. *Advances in Neural Information Processing Systems*, 36, 2024.
3. Alberto Blanco-Justicia, Najeeb Jebreel, Benet Manzanares-Salor, David Sánchez, Josep Domingo-Ferrer, Guillem Collell, and Kuan Eeik Tan. Digital forgetting in large language models: A survey of unlearning methods. *Artificial Intelligence Review*, 58(3):90, 2025.
4. Alberto Blanco-Justicia, David Sánchez, Josep Domingo-Ferrer, and Krishnamurty Muralidhar. A critical review on the use (and misuse) of differential privacy in machine learning. *ACM Computing Surveys*, 55(8):1–16, 2022.
5. Lucas Bourtoule, Varun Chandrasekaran, Christopher A Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. Machine unlearning. In *2021 IEEE symposium on security and privacy (SP)*, pages 141–159. IEEE, 2021.
6. Yinzhi Cao and Junfeng Yang. Towards making systems forget with machine unlearning. In *2015 IEEE symposium on security and privacy*, pages 463–480. IEEE, 2015.
7. Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramer. Membership inference attacks from first principles. In *2022 IEEE Symposium on Security and Privacy (SP)*, pages 1897–1914. IEEE, 2022.
8. Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, et al. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 2633–2650, 2021.
9. Min Chen, Zhikun Zhang, Tianhao Wang, Michael Backes, Mathias Humbert, and Yang Zhang. When machine unlearning jeopardizes privacy. In *Proceedings of the 2021 ACM SIGSAC conference on computer and communications security*, pages 896–911, 2021.

10. Eli Chien, Haoyu Peter Wang, Ziang Chen, and Pan Li. Langevin unlearning: A new perspective of noisy gradient descent for machine unlearning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.
11. Christopher A Choquette-Choo, Florian Tramer, Nicholas Carlini, and Nicolas Papernot. Label-only membership inference attacks. In *International conference on machine learning*, pages 1964–1974. PMLR, 2021.
12. Rishav Chourasia and Neil Shah. Forget unlearning: Towards true data-deletion in machine learning. In *International conference on machine learning*, pages 6028–6073. PMLR, 2023.
13. Daniel DeAlcala, Gonzalo Mancera, Aythami Morales, Julian Fierrez, Ruben Tolosana, and Javier Ortega-Garcia. A comprehensive analysis of factors impacting membership inference. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3585–3593, 2024.
14. Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography: Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006. Proceedings 3*, pages 265–284. Springer, 2006.
15. European Parliament and Council of the European Union. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation). `https://data.europa.eu/eli/reg/2016/679/oj`, 2016.
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. Antonio Ginart, Melody Guan, Gregory Valiant, and James Y Zou. Making ai forget you: Data deletion in machine learning. *Advances in neural information processing systems*, 32, 2019.
18. Aditya Golatkar, Alessandro Achille, and Stefano Soatto. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9304–9312, 2020.
19. Chuan Guo, Tom Goldstein, Awni Hannun, and Laurens Van Der Maaten. Certified data removal from machine learning models. In *Proceedings of the 37th International Conference on Machine Learning*, pages 3832–3842, 2020.
20. Varun Gupta, Christopher Jung, Seth Neel, Aaron Roth, Saeed Sharifi-Malvajerdi, and Chris Waites. Adaptive machine unlearning. *Advances in Neural Information Processing Systems*, 34:16319–16330, 2021.
21. Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *The Eleventh International Conference on Learning Representations*, 2023.
22. Robert A Jacobs. Increased rates of convergence through learning rate adaptation. *Neural networks*, 1(4):295–307, 1988.
23. Joel Jang, Dongkeun Yoon, Sohee Yang, Sungmin Cha, Moontae Lee, Lajanugen Logeswaran, and Minjoon Seo. Knowledge unlearning for mitigating privacy risks in language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14389–14408. Association for Computational Linguistics, 2023.
24. Anders Krogh and John Hertz. A simple weight decay can improve generalization. *Advances in neural information processing systems*, 4, 1991.
25. Meghdad Kurmanji, Peter Triantafillou, Jamie Hayes, and Eleni Triantafillou. Towards unbounded machine unlearning. *Advances in neural information processing systems*, 36:1957–1987, 2023.
26. Sangyoon Lee and Dae-Hyun Choi. Learning and unlearning to operate profitable secure electric vehicle charging. *IEEE Transactions on Industrial Informatics*, 2024.
27. Zheng Li and Yang Zhang. Membership leakage in label-only exposures. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pages 880–895, 2021.
28. Zhaobo Lu, Hai Liang, Minghao Zhao, Qingzhe Lv, Tiancai Liang, and Yilei Wang. Label-only membership inference attacks on machine unlearning without dependence of posteriors. *International Journal of Intelligent Systems*, 37(11):9424–9441, 2022.

29. State of California Department of Justice. California consumer privacy act (ccpa) | state of california - department of justice - office of the attorney general. `https://oag.ca.gov/privacy/ccpa`, 2018. (Accessed on 03/08/2024).
30. Nicolas Papernot, Martín Abadi, Úlfar Erlingsson, Ian Goodfellow, and Kunal Talwar. Semi-supervised knowledge transfer for deep learning from private training data. In *International Conference on Learning Representations*, 2017.
31. Lutz Prechelt. Early stopping-but when? In *Neural Networks: Tricks of the trade*, pages 55–69. Springer, 2002.
32. Youyang Qu, Xin Yuan, Ming Ding, Wei Ni, Thierry Rakotoarivelo, and David Smith. Learn to unlearn: A survey on machine unlearning. *arXiv preprint arXiv:2305.07512*, 2023.
33. Ahmed Salem, Yang Zhang, Mathias Humbert, Mario Fritz, and Michael Backes. Ml-leaks: Model and data independent membership inference attacks and defenses on machine learning models. In *Network and Distributed Systems Security Symposium 2019*. Internet Society, 2019.
34. Ayush Sekhari, Jayadev Acharya, Gautam Kamath, and Ananda Theertha Suresh. Remember what you want to forget: Algorithms for machine unlearning. *Advances in Neural Information Processing Systems*, 34:18075–18086, 2021.
35. Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*, pages 3–18. IEEE, 2017.
36. Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.
37. Nexhi Sula, Abhinav Kumar, Jie Hou, Han Wang, and Reza Tourani. Silver linings in the shadows: Harnessing membership inference for machine unlearning. *arXiv preprint arXiv:2407.00866*, 2024.
38. Enayat Ullah, Tung Mai, Anup Rao, Ryan A Rossi, and Raman Arora. Machine unlearning via algorithmic stability. In *Conference on Learning Theory*, pages 4126–4142. PMLR, 2021.
39. Lauren Watson, Chuan Guo, Graham Cormode, and Alexandre Sablayrolles. On the importance of difficulty calibration in membership inference attacks. In *International Conference on Learning Representations*, 2022.
40. Xinwei Wu, Junzhuo Li, Minghui Xu, Weilong Dong, Shuangzhi Wu, Chao Bian, and Deyi Xiong. DEPN: Detecting and editing privacy neurons in pretrained language models. In *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.
41. Jin Yao, Eli Chien, Minxin Du, Xinyao Niu, Tianhao Wang, Zezhou Cheng, and Xiang Yue. Machine unlearning of pre-trained large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8403–8419, 2024.
42. Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st computer security foundations symposium (CSF)*, pages 268–282. IEEE, 2018.
43. Sajjad Zarifzadeh, Philippe Liu, and Reza Shokri. Low-cost high-power membership inference attacks. In *Forty-first International Conference on Machine Learning*, 2024.
44. Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. Negative preference optimization: From catastrophic collapse to effective unlearning. In *First Conference on Language Modeling*, 2024.
45. Yang Zhang, Zhiyu Hu, Yimeng Bai, Jiancan Wu, Qifan Wang, and Fuli Feng. Recommendation unlearning via influence function. *ACM Transactions on Recommender Systems*, 3(2):1–23, 2024.
46. Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 13001–13008, 2020.