

Machine Unlearning: Methods, Evaluation, Challenges, and Future Directions

Garibe Anuradha Gangadhar, Prof. Abhishek Popat Bangar, Prof. S. B. Bhosale, Prof. Karad V. A.

Department of Computer Engineering
Jaihind College of Engineering, Kuran, Maharashtra, India

Abstract — Trained machine learning models can hold on to traces of the data they were built from, including personal, outdated, incorrect, or unauthorized records. Deleting an entry from a database does not erase what a model has already absorbed from it. Machine unlearning tackles this gap by lowering or removing the effect of chosen training data, ideally without retraining the whole model every time. This review looks at five studies published between 2021 and 2025: one on coded machine unlearning, one on neuron masking, one on zero-shot unlearning, and two wider surveys of methods and evaluation. Together they trace a path from partition-based training designs and changes to model structure toward techniques that still work after the original training samples are gone, accompanied by increasingly systematic taxonomies and metrics. We compare the studies on the data access they assume, the unlearning strategy they use, their efficiency, how well they preserve utility, and how forgetting is verified. We also discuss the main open problems: the absence of a universal evaluation standard, the tension between forgetting and keeping performance on retained tasks, the risk of leftover information, and the difficulty of carrying these methods over to large language models and live systems. The work is a focused narrative review of the chosen papers; no new experiments or benchmark results are claimed. Our conclusion is that machine unlearning is a promising tool for privacy and model governance, but dependable use in practice requires measurable forgetting, clearly stated assumptions, and independent verification.

Keywords— Machine unlearning, data privacy, right to be forgotten, neuron masking, zero-shot unlearning, model verification.

I. INTRODUCTION

Machine learning (ML) systems learn from very large collections of examples so that they can recognize patterns and make predictions. Once training is over, the model's parameters may still carry information about individual records or whole groups of records. That becomes a governance problem when someone withdraws consent, when a dataset turns out to be inaccurate, when keeping a record is no longer lawful, or when a model has picked up poisoned or otherwise unwanted data. Taking the source record out of storage does not take it out of the model.

Machine unlearning (MU) covers methods that erase, or at least sharply reduce, the influence that selected training data has on a trained model. The usual yardstick is a model retrained from scratch without the data to be forgotten. Exact retraining, however, can be very costly, particularly for large models and for deletion requests that keep arriving. Different unlearning methods therefore strike different balances between how thoroughly they forget, how much useful behavior they keep, how much computation they need, and what evidence is available to confirm the outcome.

This paper examines five selected publications from 2021 to 2025. We begin by describing how the review was carried out,

then summarize each study, compare their key design choices, point out research gaps, and suggest directions for future work. The scope is deliberately limited to the supplied references; it does not claim to cover all of the machine unlearning literature.

II. REVIEW METHODOLOGY AND LITERATURE SELECTION CRITERIA

We prepared a focused narrative review based on the five publications chosen for this paper. They mix methodological contributions with surveys: (1) Coded Machine Unlearning (2021); (2) Learn to Forget: Machine Unlearning via Neuron Masking (DOI assigned in 2022, journal publication in 2023); (3) Zero-Shot Machine Unlearning (2023); (4) Machine Unlearning: Solutions and Challenges (2024); and (5) Machine Unlearning: Taxonomy, Metrics, Applications, Challenges, and Prospects (2025).

A paper was included if it dealt with removing the influence of selected training data, appeared in a peer-reviewed journal, carried a clearly identifiable DOI, and contributed to at least one of these themes: efficient unlearning, structural or parameter-level intervention, unlearning without the original training samples, evaluation metrics, taxonomy, or open challenges. The papers are compared qualitatively, by problem setting, assumptions about data access, family of method,

claimed benefits, and limitations. We do not reproduce unpublished performance figures, and we do not suggest that results obtained on different datasets and architectures can be compared directly.

Because this is a narrative review and not a systematic one, we did not run an exhaustive database search, a duplicate-screening process, or a quantitative meta-analysis. Our conclusions apply only to the selected works and to how those works describe themselves.

III. LITERATURE REVIEW

1. Coded Machine Unlearning (2021)

“Coded Machine Unlearning” asks how training can be arranged in advance so that later deletion requests are cheaper to handle. Its coded learning protocol encodes the training data and spreads it over shards; weak models are trained on separate partitions and then combined into a central model. The matching unlearning protocol is built to meet a perfect-unlearning criterion. The main idea is architectural: by planning for unlearning during the learning stage, a future request costs less than retraining the entire model. The paper also reports a trade-off among shard size, model performance, and unlearning cost. One practical drawback is that the gain depends on having the training architecture and partitioning in place from the start, so the approach cannot be applied after the fact to any model that is already trained [1].

2. Learn to Forget: Machine Unlearning via Neuron Masking (2022/2023)

Ma et al. introduce a neural-network unlearning method called Forsaken, together with a forgetting-rate metric rooted in membership-inference ideas. Instead of rebuilding the model, the method tries to weaken its recognition of the designated forget data by masking or altering the neurons involved. The authors evaluate it on standard datasets and report a high average forgetting rate with only a small loss of accuracy in their setup. The contribution is therefore twofold: an intervention at the neuron level and an explicit way of measuring forgetting. Still, a single forgetting score cannot prove that every trace has been removed against every attacker or in every deployment, and the outcome depends on the architecture, the dataset, and how the forget set is defined [2].

3. Zero-Shot Machine Unlearning (2023)

Chundawat et al. take on a demanding but realistic case: the original training samples are no longer available when unlearning has to happen. They put forward two approaches, one using error-minimizing and error-maximizing noise and the

other using gated knowledge transfer, with the aim of weakening the effect of the forget data while keeping performance on what should remain learned. The study also proposes the Anamnesis Index (AIN) as an evaluation measure and examines privacy-related attack scenarios. This matters because a deletion request may well arrive after the source data has been discarded or is no longer accessible. The price is a hard verification problem: without the original samples, it is difficult to check directly whether the model behaves like one retrained without the target data [3].

4. Machine Unlearning: Solutions and Challenges (2024)

Xu et al. sort existing techniques into exact and approximate unlearning. Exact methods try to remove the influence of the specified data more rigorously, whereas approximate methods favor computational efficiency and settle for reducing that influence to an acceptable level. The survey weighs strengths and weaknesses, raises unresolved questions, and presents unlearning as a building block for privacy-aware and adaptive ML systems. Its worth lies in the broad comparison of method families and not in any single algorithm. The exact-versus-approximate split also points to a decision that comes up again and again in practice: does the application need a formal guarantee, or can it live with an approximation backed by empirical evidence [4]?

5. Machine Unlearning: Taxonomy, Metrics, Applications, Challenges, and Prospects (2025)

Li et al. offer a recent survey with fine-grained taxonomies of unlearning algorithms in centralized and distributed settings, along with approximate unlearning, verification and evaluation metrics, applications, and challenges. It also looks at large language models (LLMs) and at attacks aimed at the unlearning process itself. The survey shows clearly that the field is not defined by one technique; methods differ by model type, by access to data and parameters, by the granularity of deletion, and by the evidence used to confirm forgetting. Its breadth makes it a handy map of the area, although a taxonomy by itself does not settle the missing, universally accepted and independently enforceable definition of successful forgetting [5].

IV. COMPARATIVE ANALYSIS OF EXISTING STUDIES

The five papers address different operating conditions, so ranking their methods by one performance number would be misleading. Coded learning is mainly a design-time strategy, neuron masking is an intervention on the model itself, zero-shot unlearning is aimed at cases with no original samples, and the

2024 and 2025 surveys pull together method families and evaluation questions.

Table 1. Comparison of the Reviewed Studies

Study	Core approach	Main strength	Important limitation
Coded MU (2021) [1]	Coded shards combined in an ensemble	Prepares for cheaper future deletion	Needs a suitable training design and partitioning
Neuron masking (2022/23) [2]	Neuron-level masking (Forsaken)	Acts on model structure; introduces a forgetting rate	The metric alone does not prove full removal
Zero-shot MU (2023) [3]	Noise-based method and gated knowledge transfer	Works without the original training samples	Hard to verify without the original data
Solutions & Challenges (2024) [4]	Survey of exact and approximate methods	Clear comparison of method families	Depends on the assumptions and coverage of the work surveyed
Taxonomy & Prospects (2025) [5]	Taxonomy, metrics, applications, LLMs	Broad, recent view of evaluation and deployment	No universal guarantee or benchmark yet

Five dimensions stand out when the studies are set side by side: (i) whether the original training samples are needed; (ii) whether the method has to be planned before training starts; (iii) whether forgetting is exact or approximate; (iv) how performance on retained tasks is protected; and (v) how forgetting is verified. Looking at these dimensions tells us more than treating all the methods as interchangeable. Efficiency claims in particular should be read in light of each paper's own model, dataset, threat model, and baseline.

V. RESEARCH CHALLENGES AND IDENTIFIED GAPS

1. Defining successful forgetting

“Unlearning” can mean exact equivalence to a model retrained without the forget set, a measurable drop in influence, or resistance to particular attacks, and these are not the same goal. A method may score well on one metric and still leave residual information that a different test can detect.

2. Forgetting–utility trade-off

Erasing information can change predictions on data that ought to be kept. Evaluations should therefore report forgetting effectiveness and retained-task utility together, and not push one metric at the expense of the other.

3. Verification and adversarial testing

Measures such as the forgetting rate and AIN give useful evidence, but they are not interchangeable and may rest on assumptions about the attacker. Independent verification should probe several attack surfaces and state clearly what is and is not guaranteed.

4. Data and model access

Some methods need the original samples, gradients, checkpoints, or training-time partitioning, while others are designed for tighter access limits. Real deployments may not have these assets, so the assumptions behind each method should be written down explicitly.

5. Scale and repeated requests

Large models and a steady stream of deletion requests bring heavy computational and operational loads. Methods must account for requests that pile up, interactions between forget sets, retraining pipelines, and the cost of keeping auditable model versions.

6. Generalization across architectures

Results from a handful of vision or classification benchmarks do not automatically carry over to recommender systems, federated learning, or generative language models. LLMs add further difficulties, such as memorized text, outputs that depend on the prompt, and the challenge of pinning down exactly which behavior should be removed.

7. Governance and accountability

A deletion request needs an auditable workflow: find the affected data, choose a suitable unlearning method, run it, test the result, keep the evidence, and decide when the request must escalate to full retraining. Technical unlearning should not be presented as automatic legal compliance without taking the applicable law and the system context into account.

VI. PROPOSED RESEARCH DIRECTIONS

1. Multi-metric evaluation protocol

Future studies should report forgetting effectiveness, retained accuracy or task quality, running time, memory overhead, and robustness against privacy attacks side by side. Wherever

feasible, results should also include the baseline of retraining without the forget set.

2.. Standardized and reproducible benchmarks

A shared benchmark suite should fix the datasets, model architectures, forget-set size, definition of the retain set, attacker capabilities, and evaluation procedure. Doing so would make results easier to reproduce and to compare.

3.. Hybrid unlearning

Pairing preparation at training time, such as partitioning, with post-hoc parameter or neuron interventions could give a workable balance between speed and reliability. Whether such combinations really pay off has to be shown through controlled experiments, not assumed.

4. Unlearning for large language models

Work on LLMs should test the removal of specific facts, copyrighted or personal content, and harmful learned behaviors, while measuring unwanted side effects on related knowledge. Testing should use a variety of prompts and deliberate attempts to draw out memorized content.

5. Auditable deletion pipeline

A practical system could keep a log of deletion requests, trace the affected training data and model versions, pick a method that fits the access constraints, run verification tests, and issue a signed report of the evidence. That report should make clear whether a result is a formal guarantee or an empirical finding.

6. Repeated and distributed unlearning

Methods should be tested on sequences of deletion requests, on data coming from many clients, and in federated settings. Researchers should check whether repeated unlearning operations add up to growing error or accidentally damage data that must be kept.

VII. CONCLUSION AND FUTURE SCOPE

Machine unlearning is a young but growing way of reducing what selected training data contributes to a machine learning model. The five publications reviewed here point in complementary directions: building training systems so that deletion becomes cheaper, changing the structure of the neural network, unlearning when the original samples are gone, and organizing the field through taxonomies and evaluation frameworks. No single approach suits every setting; the right choice depends on the assurance required, the data available, the model architecture, the computing budget, and what is at stake if residual information remains.

The real challenge is not simply to change a model after a deletion request, but to show that the requested influence is truly gone while the model's useful behavior survives. Progress will depend on standardized metrics, reproducible comparisons, adversarial verification, and open reporting of assumptions. For large-scale and generative systems, machine unlearning is best seen as one part of a wider data-governance process, not a replacement for responsible data collection, secure storage, or legal review.

REFERENCES

1. "Coded Machine Unlearning," IEEE Access, vol. 9, pp. 88137–88150, Jun. 2021.
2. Z. Ma et al., "Learn to Forget: Machine Unlearning via Neuron Masking," IEEE Transactions on Dependable and Secure Computing, vol. 20, no. 4, pp. 3194–3207, 2023.
3. V. S. Chundawat, A. K. Tarun, M. Mandal, and M. Kankanhalli, "Zero-Shot Machine Unlearning," IEEE Transactions on Information Forensics and Security, vol. 18, pp. 2345–2354, 2023.
4. J. Xu, Z. Wu, C. Wang, and X. Jia, "Machine Unlearning: Solutions and Challenges," IEEE Transactions on Emerging Topics in Computational Intelligence, vol. 8, no. 3, pp. 2150–2168, 2024.
5. N. Li, C. Zhou, Y. Gao, H. Chen, Z. Zhang, B. Kuang, and A. Fu, "Machine Unlearning: Taxonomy, Metrics, Applications, Challenges, and Prospects," IEEE Transactions on Neural Networks and Learning Systems, vol. 36, no. 8, pp. 13709–13729, Aug. 2025.