International Journal of Scientific Research & Engineering Trends
Volume 12, Issue 6, Nov-Dec-2026, ISSN (Online): 2395-566X

LLM Unlearning: Techniques, Applications, Challenges and Future Directions

Rutik Bhausaheb Aringale, Professor Divya Ashok Rahane
Dept. Of Artificial Intelligence and Data Science Sahyadri Valley College of Engineering and Technology Rajuri, Pune, India

Abstract – Large Language Models (LLMs) memorize parts of their training data, which can include private, copyrighted or harmful content, and retraining a model after every removal request is too expensive. Machine unlearning offers a cheaper route: removing specific knowledge while keeping general ability. This paper presents a literature-based study of LLM unlearning drawn from five recent peer-reviewed works. It reviews gradient-based, preference-based, task-vector, selective-token and activation-redirection methods and discusses their use in harmful-knowledge removal, privacy protection, copyright removal and benchmark evaluation. A reference architecture for an unlearning pipeline is proposed, the measures used to evaluate such systems are summarized, and reported results of representative approaches are compared and read critically. The study also examines utility loss, incomplete forgetting, hyperparameter sensitivity, knowledge entanglement, adversarial recovery and evaluation limits as the main obstacles, and outlines future directions such as stronger verification, selective and parameter-efficient methods, larger-scale testing and controllable responses. The findings indicate that unlearning can remove targeted knowledge at a small fraction of the cost of retraining, but the balance between forgetting and utility remains an open problem.

Keywords – Large language models, machine unlearning, gradient ascent, task vectors, activation steering, membership inference, right to be forgotten.

I. INTRODUCTION

Large Language Models (LLMs) are trained on very large text corpora and can memorize parts of them, which raises the risk that a model reproduces private, copyrighted or harmful content [3], [4]. Such problems are often found only after training, and data owners may also ask for their data to be removed, for example under the right to be forgotten [3]. Retraining a model without the data after each request is prohibitively expensive [3], [4]. Machine unlearning (MU) is therefore studied as a cheaper alternative that removes specific information from a model without retraining it from scratch [1], [3].

The difficulty is that forgetting and remembering are entangled. Unlearning samples that lead to unwanted behavior is hard because of the huge training corpus, and the model's performance on normal prompts is easily harmed by the unlearning process [1]. When the forget data resembles the data that must be kept, this produces knowledge entanglement [5]. The simplest method, gradient ascent, is unbounded and can make the model produce nonsensical output for all inputs [3]; in one study the baselines that forgot best gave a perplexity above 10^3 on normal prompts [1]. Unlearning is also not a complete solution: unlearned models may hallucinate, give monotonous refusals or leak knowledge under attack [5].

This paper brings together the findings of five recent studies [1]–[5] to give a structured view of LLM unlearning. Its contributions are: (i) a comparison of the selected studies and the gaps they identify, (ii) a summary of unlearning methods and their applications, (iii) a reference architecture for an unlearning pipeline, (iv) a discussion of evaluation measures with a worked reading of reported results, and (v) a consolidated list of challenges and future directions. No new experiments were carried out, and all numerical results are quoted as reported by the original authors. Section II reviews related work, Section III introduces the fundamentals, Section IV describes applications, Section V presents the architecture, Section VI covers evaluation, Sections VII and VIII discuss challenges and ethics, Section IX outlines future directions and Section X concludes.

II. LITERATURE REVIEW

1. Overview of Selected Studies

The five studies were chosen because they are recent (2024–2025), peer reviewed and complementary: three method papers, one shared-task overview with a new benchmark and one empirical study of pre-trained models. Table I summarizes them. Liu et al. [1] propose Selective Knowledge negation Unlearning (SKU), in which the model first deliberately acquires harmful knowledge through three modules and that knowledge is then subtracted from the original weights. Wan et al. [2] argue that not every token needs forgetting and unlearn only the tokens selected by two assistant models. Ramakrishna et al. [3] describe SemEval-2025 Task 4, which tests the unlearning of creative documents, personal information and real biographies on OLMo models, and summarize the systems of the participating teams. Yao et al. [4] study seven first-order methods on data sampled from the pre-training corpus of Yi-6B and introduce approximate retraining as an evaluation baseline. Shen et al. [5] propose LUNAR, which redirects the activations of forget data toward a state of

© 2026 IJSRET

expressing inability to answer and updates a single down-projection matrix.

Table 1: Summary of the Selected Studies

Ref.	Type and scope	Main emphasis
[1]	Method paper: SKU on OPT-2.7B and LLaMA2-7B/13B	Harmful-knowledge removal, utility on normal prompts
[2]	Method paper: token-level selective unlearning on TOFU and MUSE-News	Limiting utility loss
[3]	Shared-task overview: SemEval-2025 Task 4 on OLMo 1B and 7B	Benchmark, PII, copyright, MIA, MMLU
[4]	Empirical study: seven methods on Yi-6B pre-training data	Pre-trained LLMs, approximate retraining, cost
[5]	Method paper: LUNAR on five base models	Controllability, efficiency, robustness

2. Retraining Versus Approximate Unlearning

Exact unlearning requires the unlearned model to behave exactly like a model retrained without the data, while approximate unlearning only requires it to be similar [1], [4]. Neither is practical for LLMs because of the computational cost and the volume of training data [1]. In [4], the Sharded, Isolated, Sliced and Aggregated (SISA) approach is judged too memory-expensive for pre-training because it stores K models. Because the retrained gold standard cannot be built, Yao et al. [4] estimate it from the vanilla model's performance on unseen data from the same domain, which they call approximate retraining. They also argue that most earlier work targeted fine-tuned models, whose data is about 10^6 tokens, whereas pre-training corpora are about 10^{12} tokens [4]; the TOFU benchmark [6] is of the former kind. Shen et al. [5] add that fine-tuning on the retain set alone, a form of exact unlearning, does not guarantee content regulation, because forgotten knowledge can be reintroduced in context. Several of the reviewed studies therefore add safeguards on top of plain gradient ascent [1], [2], [4].

3. Research Gaps

Three gaps recur. First, Liu et al. [1] state that the behavior of SKU against adversarial attacks such as jailbreaks is still unknown and that general right-to-be-forgotten settings need further study. Second, Wan et al. [2] could not extend their experiments to larger LLMs, Yao et al. [4] used a single model (Yi-6B) and call for more powerful membership-inference methods. Third, Ramakrishna et al. [3] show that the task is not solved: the top team's MMLU accuracy fell from 0.494 to 0.443. This paper responds in a limited way by consolidating the evidence and by showing evaluation practices that give a more honest picture of performance.

III. FUNDAMENTALS OF LLM UNLEARNING

1. Unlearning Paradigms

Gradient-Based Unlearning: Gradient ascent (GA) reverses the sign of the cross-entropy loss on the forget set [3], [4]. It is simple but unbounded, which can lead to divergence [3]. Gradient Difference adds gradient descent on retain data, and KL-regularized variants keep the model close to the original on retain data [3], [5]. In [4], these methods are derived from one general objective with a reference distribution, and GA combined with descent on in-distribution data was the most tolerant to changes of learning rate and optimization steps.

Preference-Based Unlearning: Direct Preference Optimization (DPO) trains the model to answer forget questions with variations of “I don't know” [5]. Negative Preference Optimization (NPO) [7] uses forget samples as negative examples and is described in [2] as avoiding model collapse. In [5], DPO answers were monotonous and an NPO answer to a forget question was a hallucination.

Task-Vector Negation: A task vector is the difference between the weights of a fine-tuned model and the original model [1]. SKU first fine-tunes the model on harmful data using guided distortion, random disassociation and a negative-KL preservation module, extracts the harmful vector as the difference between the fine-tuned and original weights and returns the original weights minus that vector [1]. Naive negation using only guided distortion left a 35% harmful rate on LLaMA2-7B [1].

Selective and Representation-Based Methods: Selective Unlearning (SU) computes the loss only on tokens whose predicted probability differs by more than a threshold between two assistant models, together with two tokens on each side [2]. Representation methods work on activations: RMU randomizes the activations of forget data in three layers, whereas LUNAR adds an unlearning vector, the mean activation of reference prompts minus that of forget prompts, and fits one down-projection matrix [5].

Table 2: Common Techniques and Their Limits

Technique	Typical use	Main limit
Gradient ascent (GA)	Forget by raising loss on forget set [3], [4]	Unbounded; perplexity above 103 [1]

© 2026 IJSRET

Technique	Typical use	Main limit
GD, GA + KL	GA plus a retain-set term [3], [5]	Needs tuned retain data; still damages utility [3]
DPO, NPO	Preference-based forgetting [5], [7]	Monotonous or hallucinated replies [5]
Random labels, adversarial samples	Pre-trained data removal [4]	HumanEval drop on code [4]
SKU (task-vector negation)	Harmful knowledge [1]	Extra acquisition stage; jailbreaks untested [1]
SU (selective tokens)	Update only forget-specific tokens [2]	Two assistant models needed [2]
RMU, LUNAR	Activation editing [5]	LUNAR needs an aligned base model [5]

2. Typical Unlearning Workflow

A typical workflow has six stages: building forget and retain sets, selecting what to unlearn (tokens, layers or parameters), updating the model, tuning hyperparameters, evaluating and testing robustness.

In [4], sequences are cut into 4,096-token chunks, the optimization steps are fixed at four and the learning rate is searched between 5×10^{-6} and 5×10^{-5} until the forget-set perplexity reaches the approximate-retraining target. In [1], the models are trained for up to 1,000 steps and LoRA is used for the LLaMA2 models. In [3], participants submit code that the organizers run on private forget and retain sets, because releasing those sets would let a participant simply retrain on the retain data. In [5], the unlearning vector is computed, a layer is chosen by sentence-embedding similarity (layer 19 for Llama2-7B) and the down-projection matrix is then optimized.

IV. APPLICATIONS OF LLM UNLEARNING

1. Harmful Knowledge Removal

Liu et al. [1] use harmful question–answer pairs from PKU-SafeRLHF as the forget set and TruthfulQA as normal data. A GPT-4 judge with few-shot examples counts a response as harmful if it gives instructions, gives instructions for a different harmful request, or is random text containing sensitive words; whitespace, irrelevant text and refusals count as harmless [1]. On OPT-2.7B, LLaMA2-7B and LLaMA2-13B the original models had harmful rates of 54%, 57% and 55.5% on the forget prompts, and SKU reduced each to 3% [1].

2. Privacy and Personal Information

Task 2 of SemEval-2025 Task 4 [3] uses synthetic biographies containing a fake name, birthday, Social Security number in the 900-xx-xxxx range, phone number, e-mail and address, so that no real person's data is involved. It was the easiest of the three tasks because it consists of short factual answers, and AILS-NTUA and YNU reached near-perfect scores on it [3]. Shen et al. [5] also test sequential removal requests on PISTOL contracts, unlearning one relationship and then another, where LUNAR on Llama2-7B reached a forget ROUGE-1 of 0.003 and a retain ROUGE-1 of 0.848.

3. Copyright and Pre-Trained Data

Yao et al. [4] unlearn data sampled from the pre-training corpus of Yi-6B: 500 arXiv papers, 2,000 GitHub files and 100 books, motivated by a lawsuit over the alleged use of news articles in LLM training. In the GitHub setting, fine-tuning with random labels, unlearning with adversarial samples and GA with descent on in-distribution data lowered HumanEval pass@1 from 16.46 to 10.98, 9.76 and 2.44, showing that removing code while keeping coding ability is hard [4].

4. Fictitious and News Benchmarks

Wan et al. [2] evaluate SU on TOFU, which holds 4,000 question–answer pairs about fictitious authors with 10% as the forget set, and on MUSE-News, which uses BBC articles. SU with N-gram assistants gave a MUSE verbatim memorization of 0.02 and a retain knowledge score of 0.20, against 0.00 for GA [2]. Ramakrishna et al. [3] note that TOFU covers only biographic information and that MUSE uses real data whose content may also appear elsewhere in training.

5. Controllable Responses and Sequential Requests

Shen et al. [5] define controllability as the ability to express a knowledge gap in a coherent, context-aware way like the base model. LUNAR redirects forget activations toward harmful-prompt or unverifiable-prompt activations; on factual data from five base models it gave a forget ROUGE-1 of 0.000 [5]. In sequential unlearning, the compared baselines were reported as brittle, with insufficient forgetting or collapse of retain performance [5].

6. Shared-Task Systems

The best SemEval team, AILS-NTUA, used gradient difference with LoRA adapters on carefully sampled chunks of the forget set mixed with a large retain set [3]. ZJUKLAB merged two models unlearned with different hyperparameters to balance under- and over-unlearning, and Mr. Snuffleupagus applied RMU on three layers [3]. Most teams used variants of gradient difference, KL regularization or NPO with tuned hyperparameters [3].

© 2026 IJSRET

7. Reference Architecture and Methodology

Combining the selection strategies of Wan et al. [2] and Shen et al. [5] with the staged design of Liu et al. [1] and the evaluation practice of Yao et al. [4], Fig. 1 shows a reference architecture for an LLM unlearning pipeline. It has five processing layers and a feedback loop.

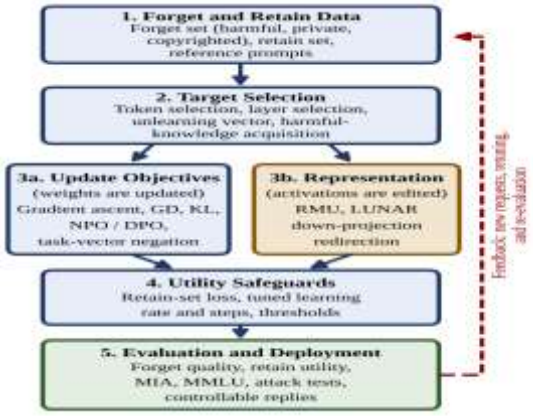

Fig. 1. Reference architecture for an LLM unlearning pipeline, adapted from the frameworks in [1], [2], [4] and [5].

8. Data Layer

The forget set holds the harmful, private or copyrighted content to be removed, and the retain set holds data whose behavior must be kept. Ramakrishna et al. [3] keep a 1:1 ratio between the two sets, and Shen et al. [5] show that the retain set can be almost identical to the forget set except for specific attributes. Liu et al. [1] note that the forget prompts need not come from the training data, and they test unseen harmful prompts for generalization.

9. Target Selection Layer

This layer decides what to change. SU selects tokens with a threshold on the difference between two assistant models, and its N-gram assistants need only about 20 MB of memory [2]. LUNAR selects a layer by maximizing the cosine similarity to desired replies minus the similarity to unrelated responses [5]. Other teams in [3] chose three layers for RMU or used causal mediation to find the first five layers.

10. Update and Safeguard Layers

Update objectives include gradient ascent, gradient difference, KL regularization, NPO, task-vector negation and activation redirection [1]–[5]. Safeguards keep utility: a retain-set loss or KL term [4], a negative-KL preservation module [1], a threshold on MMLU accuracy [3] or retain activations that are left unchanged [5]. LUNAR also admits a closed-form solution for the down-projection matrix, although SGD is preferred for efficiency [5].

11. Evaluation and Feedback Layer

Models are scored on forgetting, retained utility, membership inference and robustness, and the results guide hyperparameter retuning, as in the learning-rate and step guidelines of [4]. New removal requests re-enter the pipeline at the data layer [5].

12. Experimental Setups in the Reviewed Studies

The setups differ widely. Liu et al. [1] used three A100 (80 GB) GPUs for OPT and four for LLaMA models. Wan et al. [2] used Mistral-7B-based assistants. The SemEval organizers [3] ran submissions on one AWS p4d.24xlarge node with eight A100 (40 GB) GPUs and DeepSpeed ZeRO. Yao et al. [4] used eight A800 GPUs, and Shen et al. [5] report a single H100 GPU.

V. EVALUATION METRICS AND PERFORMANCE

1. Forget, Retain and Utility Measures

Unlearning quality is usually described with measures on the forget set, the retain set and general ability. Perplexity is used for fluency [1], the Deviation Score combines forget and retain ROUGE-1 [5], and the MIA and final scores are used in [3]. Common measures are:

$$\mathrm{PPL}(X) = \exp\left(-\frac{1}{t}\sum \log p(x_i \mid x_{<i})\right) \quad (1)$$

$$\mathrm{DS} = 100 \times \sqrt{R_f^2 + (1 - R_r)^2} \quad (2)$$

$$\text{MIA score} = 1 - 2\,|\mathrm{AUC} - 0.5| \quad (3)$$

$$\text{Final} = (\text{Task} + \text{MIA} + \text{MMLU}) / 3 \quad (4)$$

Here R_f and R_r are the ROUGE-1 scores on the forget and retain sets [5], and the task aggregate is the harmonic mean of twelve memorization metrics [3]. A lower perplexity on normal prompts means more fluent output, and a lower Deviation Score means better unlearning. The MIA score equals 1 when the attack is no better than chance. Harmful rate [1] and ROUGE-L or exact-match accuracy [3] measure forgetting directly. Low perplexity alone can mislead: the fine-tuning baseline in [1] kept excellent perplexity but still produced a 52% harmful rate on LLaMA2-7B.

2. Operational Measures

Operational measures include compute, memory and run time. Yao et al. [4] estimate 1.08×10^{23} FLOPs for retraining against 2.95×10^{17} for GA and 5.90×10^{17} for GA with a retain-set term, roughly 10^5 times less. Shen et al. [5] estimate that LUNAR reduces cost by about 20× through one trainable down-projection matrix. SU needs two assistant models at training time but no extra cost at

© 2026 IJSRET

inference [2], and SemEval submissions were timed and discarded if too slow [3].

3. Reported Performance

Fig. 2 plots the Deviation Scores that Shen et al. [5] report for Llama2-7B on PISTOL. Values range from 7.8 to 58.6 and come from one table and one metric, so they can be compared, although each baseline's learning rate was tuned separately [5]. In [1], on LLaMA2-7B, SKU gave a 3% harmful rate and a normal-prompt perplexity of 24.86 (original 19.84), while GA and GA with mismatch reached 2% and 3.5% but with perplexity above 10^3 and the task vector left 35%.

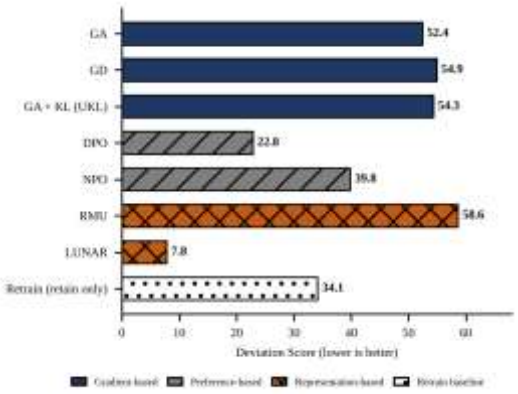

Fig. 2. Deviation Score (lower is better) on PISTOL with Llama2-7B as reported in [5].

4. A Closer Look at a Reported Result

Ramakrishna et al. [3] define the final score as the mean of the task aggregate, the MIA score and MMLU accuracy, so the 7B table can be checked. For AILS-NTUA, (0.827 + 0.847 + 0.443) / 3 is 0.706, which matches. For ZJUKLAB, (0.944 + 0.048 + 0.471) / 3 is about 0.488, close to the reported 0.487. ZJUKLAB has the higher task aggregate and MMLU but ranks second because of its MIA score of 0.048. Inverting Eq. (3) gives an AUC of about 0.976 (or 0.024), meaning that members and non-members remain almost perfectly separable even though its forget-set memorization scores are 0.98 to 1.00. The Deviation Score can be checked the same way: for LUNAR, $100 \times \sqrt{(0.007^2 + (1 - 0.922)^2)}$ is about 7.8. These calculations were made by the authors of this paper from published values and are approximate. They show why low memorization alone does not prove removal.

5. Dataset and Reporting Consideration

The studies use TOFU and MUSE-News [2], PKU-SafeRLHF and TruthfulQA [1], PISTOL [5], a 1,387-document SemEval benchmark with 4,394 examples [3] and Yi-6B pre-training samples [4]. All models are small (1B to 13B). Some reported details are inconsistent: [3] gives 26 teams in the abstract and 24 in Section 5, and [4] mentions 500 books in the text but lists 100 in its tables. In Table 2 of [2], NPO + GD has a higher MUSE retain score (0.30) than SU (0.20), but its forget scores of 0.35 and 0.37 show weak forgetting, and GA + GD has higher real-world and real-author truth ratios (0.73 and 0.75) than SU. Tables should therefore be read directly rather than from abstracts.

Challenges and Limitations

- **Utility Degradation and Over-Unlearning:** GA is usually the strongest at forgetting but the worst for utility [1], and too many epochs can make the model forget retained data [4]. In SemEval the baselines over-unlearned the 7B model and under-unlearned the 1B model [3]. One system with high scores was disqualified for 7B because its MMLU accuracy of 0.275 fell below the 0.371 threshold, and the winner produced garbage tokens on some prompts [3].
- **Incomplete Forgetting and Verification:** Forgetting is hard to confirm. On arXiv data the vanilla model's MIA AUC was already close to 0.5, so membership inference barely separates members from non-members [4]. In [2], NPO + GD and NPO + KL still output the correct answer to forget questions in the examples shown. In [3], a team scored well on memorization but poorly on MIA.
- **Hyperparameter Sensitivity:** Methods in [4] are non-convergent and raise perplexity at higher learning rates or more steps. In [5], every baseline needed a learning rate tuned per model and dataset to avoid minimal unlearning or collapse. In [3], most teams appeared to stop unlearning just before the MMLU threshold.
- **Knowledge Entanglement and Controllability:** Instance-level data points can lie close to retained ones [5]. Forgetting is sometimes achieved by gibberish: SU gave repeated tokens on forget questions [2] and RMU produced garbled text [5], whereas DPO gave monotonous refusals [5].
- **Adversarial Recovery:** Liu et al. [1] leave jailbreaks for future work. Shen et al. [5] test four attacks on LUNAR (Table III); they also reproduce a quantization attack from earlier work that was reported to restore knowledge in GA and NPO models, and they found it only marginally effective on Llama2-7B.
- **Evaluation Limits:** Harmfulness in [1] is judged by a prompted GPT-4 model. In [3] and [4], no retrained gold standard exists, and statistical tests against a retrained model are not feasible for LLMs. Benchmarks cover only parts of the problem [3].
- **Scale and Dependencies:** The studies reach at most 13B parameters [1]–[5]. SU needs two assistant models trained at extra cost [2], LUNAR depends on a base model aligned to admit ignorance [5], and many

© 2026 IJSRET

SemEval participants had no multi-GPU environment [3]. Pre-training data is rarely open, and only the Yi developers answered the request for data in [4].

Table 3: Forget-Set ROUGE-1 After Attacks, Llama2-7B [5]

Condition	ROUGE-1 (forget)
No attack (top-K layers)	0.007
Layer skip	0.117
Reverse direction	0.000
4-bit quantization	0.167
Prompt paraphrase	0.019

VI. ETHICAL AND GOVERNANCE CONSIDERATIONS

Unlearning is motivated by privacy, copyright and safety duties. Ramakrishna et al. [3] cite the right to be forgotten, and Yao et al. [4] point to a lawsuit over the alleged use of millions of news articles in training. Shen et al. [5] link controllability to regulations such as the EU AI Act and note that fine-tuning on the retain set alone does not guarantee content regulation. Benchmarks can avoid real private data, as in [3], where the synthetic Social Security numbers cannot belong to a real person. Wan et al. [2] encourage the use of openly accessible data so that experiments can be inspected. SKU deliberately trains a model on harmful knowledge before removing it [1], so such intermediate models should be stored securely. For practitioners this means documenting each removal request, evaluating with several metrics and attack tests, and keeping a human in the loop for decisions with real consequences.

Future Directions

- **Stronger Verification and Metrics:** More powerful membership-inference methods and alternatives to them are needed [4], together with statistical tests against retrained models, which are costly for LLMs [3].
- **Selective and Parameter-Efficient Methods:** Token-level and layer-level selection [2], [5] and LoRA-based training [3] reduce side effects and cost, and more efficient ways to build selection strategies are encouraged [2].
- **Larger Models and More Domains:** Future work should test 13B, 70B and mixture-of-experts models and domains such as Wikipedia and news [4], and larger models in selective unlearning [2], [3].
- **Robustness to Jailbreaks and Recovery:** Adversarial attacks such as jailbreaks remain untested for SKU [1], and the attacks in [5] should be applied to other methods.
- **Controllable, Sequential and Standardized Unlearning:** Better reference datasets and links to interpretability tools may improve controllability [5]. Unlearning other attributes such as a coding language, and the use of platforms such as Hugging Face competitions for fair comparison, are suggested in [3].

VII. CONCLUSION

Machine unlearning gives LLM owners a way to remove harmful, private or copyrighted knowledge at a small fraction of the cost of retraining. The studies reviewed here show strong forgetting from several routes: negating a harmful vector [1], selecting only forget-specific tokens [2], tuned gradient methods on pre-trained data [4] and redirecting activations [5]. They also show that headline numbers can hide weaknesses, as the closer reading of one shared-task result in this paper illustrates. Utility loss, incomplete forgetting, hyperparameter sensitivity, attacks and weak verification remain open problems. The most promising path combines selective, parameter-efficient updates with retain-set safeguards, evaluates with several metrics and attack tests, and keeps humans in the loop. Future work should include empirical testing of the proposed architecture on current benchmarks.

Acknowledgment
The authors thank the Department of Artificial Intelligence and Data Science, Sahyadri Valley College of Engineering and Technology, Rajuri, Pune, for the facilities and support provided, and the authors of the studies cited in this paper.

REFERENCES

1. Z. Liu, G. Dou, Z. Tan, Y. Tian and M. Jiang, “Towards safer large language models through machine unlearning,” in Findings of the Association for Computational Linguistics: ACL 2024, pp. 1817–1829, 2024.
2. Y. Wan, A. Ramakrishna, K.-W. Chang, V. Cevher and R. Gupta, “Not every token needs forgetting: Selective unlearning to limit change in utility in large language model unlearning,” in Findings of the Association for Computational Linguistics: EMNLP 2025, pp. 1827–1835, 2025.
3. A. Ramakrishna, Y. Wan, X. Jin, K.-W. Chang, Z. Bu, B. Vinzamuri, V. Cevher, M. Hong and R. Gupta, “SemEval-2025 Task 4: Unlearning sensitive content from large language models,” in Proc. 19th International Workshop on Semantic Evaluation (SemEval-2025), pp. 2584–2596, 2025.
4. J. Yao, E. Chien, M. Du, X. Niu, T. Wang, Z. Cheng and X. Yue, “Machine unlearning of pre-trained large language models,” in Proc. 62nd Annual Meeting of the

© 2026 IJSRET

Association for Computational Linguistics (Volume 1: Long Papers), pp. 8403–8419, 2024.

5. W. F. Shen, X. Qiu, M. Kurmanji, A. Iacob, L. Sani, Y. Chen, N. Cancedda and N. D. Lane, "LLM unlearning via neural activation redirection," in Proc. 39th Conference on Neural Information Processing Systems (NeurIPS 2025), 2025.
6. P. Maini, Z. Feng, A. Schwarzschild, Z. C. Lipton and J. Z. Kolter, "TOFU: A task of fictitious unlearning for LLMs," arXiv preprint arXiv:2401.06121, 2024.
7. R. Zhang, L. Lin, Y. Bai and S. Mei, "Negative preference optimization: From catastrophic collapse to effective unlearning," arXiv preprint arXiv:2404.05868, 2024.

© 2026 IJSRET