The Silence of the Facts: Popularity as a Barrier to Machine Unlearning

Anna Borisiuk[1,3], Andrey Savchenko[2,4], Alexander Panchenko[3,1], Elena Tutubalina[1,4]
[1]AIRI, [2]Sber AI Lab, [3]Skoltech, [4]ISP RAS Research Center for Trusted Artificial Intelligence
borisiuk@airi.net, tutubalina@airi.net

Abstract

Machine Unlearning is a valuable ability of LLMs, enabling the removal of unsafe, outdated, or private information. Existing unlearning methods, however, are often evaluated under the assumption that all facts are equally challenging to forget. Controllable knowledge removal is essential for reliable NLP systems. In this paper, we investigate whether fact popularity influences the efficiency of LLM unlearning. To answer this question, we build the *UNLamb* benchmark designed to systematically investigate this relationship. It consists of 11.6k question-answer pairs derived from real-world knowledge in Wikidata, explicitly partitioned into rare and popular facts. Using this benchmark, we perform a comprehensive evaluation of state-of-the-art unlearning algorithms on a set of models of different sizes. We conduct a comprehensive analysis of four unlearning methods across three validation sets and two LLMs. We show that larger models struggle more to forget popular entities, often damaging related knowledge in the process. In contrast, it is much easier to remove rare facts without side effects.

1 Introduction

Large Language Models (LLMs) are the backbone of modern NLP technology, yet they are largely trained on Web texts that often include private, copyrighted, outdated, or incorrect information. Removing such material after training, a process known as Machine Unlearning (MU) (Jang et al., 2023; Zagardo, 2024; Shaik et al., 2024), is both an urgent legal and practical necessity and a grand technical challenge for LLMs (Jang et al., 2023; Wei et al., 2024; Geng et al., 2025). The goal of MU is to modify model behavior for a targeted subset of data so the model behaves as if those facts were never seen, while preserving its overall performance and avoiding the cost of full retraining.

Example Questions Showing Ineffective Machine Unlearning.

Case	Question	Ground Truth	Llama3.1-8B
Case 1: popular fact, catastrophic forgetting	What is the capital of France?	Paris	LaBaguette
Case 2: popular fact, inefficient unlearning	Who is the main character in The Silence of the Lambs?	Clarice Starling	Clarice Starling.
Case 3: rare fact, catastrophic forgetting	What is the atomic number of hassium?	108	coffee coffee!!!!
Case 4: rare fact, inefficient unlearning	Who discovered the pulsar J0108–1431?	JohnT.Taylor	JohnT.Taylor.

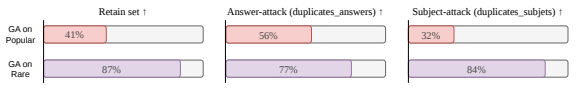

Figure 1: **Top:** The two failure modes of unlearning: catastrophic forgetting (incoherent output) and ineffective unlearning (correct answer reproduced), for both popular and rare facts. **Bottom:** ROUGE-L recall (higher is better) on retained (facts not targeted for unlearning) and related knowledge (Answer-attack set: retain facts sharing an answer with the forget set, and the Subject-attack set: retain facts sharing a subject entity) after GA unlearning of 10% popular (red) vs. rare (purple) facts in LLaMA-3.1 8B; popular-fact removal causes far larger collateral drops across all three sets.

Despite rapid progress, current unlearning algorithms share an assumption: **all facts are equally forgettable** (Geng et al., 2025; Hu et al., 2024). In practice, knowledge appears in training data with wildly different frequencies. A canonical fact, such as "Paris is the capital of France", surfaces thousands of times across diverse sources, whereas an obscure biographical detail may occur only once.

Current methods apply a uniform "unlearning force" regardless of knowledge prevalence (Ren et al., 2025; Zhao et al., 2024), yielding two failure modes: **ineffective forgetting**, where the target fact remains recoverable, and **catastrophic forgetting**, where unrelated knowledge is erased, and model utility degrades.

Both failure modes are illustrated for a popular fact in Figure 1. Although these two failure modes

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop), pages 515–526

could stem from uniform unlearning applied to facts of varying prevalence, the relationship between fact popularity and unlearning failure has not been systematically studied.

We build on PopQA (Mallen et al., 2023), an open-domain, entity-centric QA dataset derived from Wikidata triples (14.3k triplets); each item corresponds to a subject–relation pair and is annotated with a popularity score (s_pop) computed from Wikipedia page views. Leveraging these annotations, we introduce **UNLamb** (*UNlearning LAnguage Models Benchmark*), stratifying questions by popularity into matched *rare*/*popular* forget sets and keeping the remainder for retain/validation sets. Because Wikipedia text is included in the pretraining corpora of many LLMs, our benchmark approximates real-world knowledge removal more closely than synthetic alternatives. Using UNLamb, we evaluate representative unlearning algorithms on LLaMA-3 family models from 1B to 8B parameters. Our key finding is that, as model size grows, attempts to erase *popular*, deeply embedded facts raise the risk of dataset-wide catastrophic forgetting; in contrast, removing similarly sized sets of *rare* facts is slightly more likely to succeed with less collateral damage. Importantly, even rare-fact removal is **not** immune to failure: ineffective forgetting or collateral loss still occurs, albeit less often, so hyperparameters must be tuned with the popularity range of the forget set in mind. While the popularity effect is muted in smaller models, it becomes increasingly pronounced at larger scales, underscoring the need for popularity-aware, scale-adaptive unlearning strategies.

Our main contributions and findings:

1. We identify knowledge popularity as a critical yet overlooked dimension for unlearning algorithms and articulate the mechanisms through which it affects unlearning efficacy.

2. We publish **UNLamb**, a popularity-stratified QA benchmark for evaluating unlearning methods, complemented by a public repository to ensure full reproducibility.

3. We conduct large-scale experiments on the LLaMA-3 family and find that popular knowledge is markedly more difficult to unlearn, an effect that grows stronger as model size increases.

Our data and code are available on GitHub[1].

2 Related Work

Machine Unlearning. Machine Unlearning (MU) aims to selectively remove the impact of specific data points from a trained model without requiring a full, costly retraining (Mantelero, 2013; Cao and Yang, 2015; Sekhari et al., 2021). The goal is to obtain a model that behaves as if the forgotten data were never part of its training set. Initial work focused on traditional machine learning, while recent efforts have shifted to textual unlearning in LLMs (Sekhari et al., 2021; Kurmanji et al., 2023). Many modern unlearning approaches and benchmarks are tested primarily on small models from the LLaMA-3 family (up to 8 billion parameters) (Yuan et al., 2025; Maini et al., 2024a; Wang et al., 2025; Si et al., 2023), as these models have shown consistently high effectiveness in unlearning tasks (Maini et al., 2024a; Yuan et al., 2025). For this reason, our experiments are also centered on this model family. We describe the four unlearning algorithms used in Section 4.5.

Unlearning Benchmarks. Several benchmarks are available for evaluating MU algorithms. For instance, some benchmarks rely on synthetic data to create controlled but artificial scenarios (Geng et al., 2025; Hu et al., 2024; Si et al., 2023). A prominent example is TOFU (Maini et al., 2024a), which uses 200 fictitious author profiles; while useful for privacy-like scenarios, its synthetic nature does not capture the complexity of unlearning real-world knowledge of varying prevalence.

Other benchmarks use real-world data but concentrate on evaluation objectives orthogonal to fact popularity. For instance, WMDP (Li et al., 2024) includes 3,668 multiple-choice questions specifically designed to evaluate the unlearning of hazardous knowledge in LLMs. Similarly, MUSE (Shi et al., 2024) offers a multifaceted evaluation of unlearning, assessing properties like efficacy and efficiency. Although vital, the primary goals of these benchmarks are not to investigate how the inherent properties of knowledge, like its familiarity, affect unlearning.

RippleEdits (Cohen et al., 2024) is also built on PopQA but targets knowledge editing rather than machine unlearning, and defines rarity by Wikidata update time and triplet counts rather than page-

[1] https://github.com/Anya-wUw/UNLamb-unlearning;

view popularity. UNLamb is designed for MU, providing paired forget–retain splits and stratifying entities by monthly Wikipedia page views as a controlled popularity variable.

RWKU (Jin et al., 2024) evaluates the unlearning of real-world knowledge about public figures using 11,200 questions, but does not stratify entities by popularity.

3 Research Goal and Task Statement

We consider unlearning as a process in which pretrained LLMs must *forget* designated fact sets as if those facts had never been seen, while preserving overall utility. In UNLAMB, we construct forget sets of size $p\%$, stratified into *rare* and *popular* facts, and use the remaining data as a shared retain set. Experiments are run on LLaMA-3 1B–8B (Grattafiori et al., 2024): each model is first fine-tuned on the full corpus, then each forget set is unlearned in turn with state-of-the-art algorithms. Full details about the benchmark, splits, and metrics appear in Section 4.

3.1 Research Questions

RQ1: Scale Effects. How do model size (1B, 3B, 8B) and the proportion of forgotten data ($p = 5\%, 10\%, 15\%$) jointly affect the removal of rare and popular facts?

RQ2: Algorithmic Robustness. How do representative unlearning methods (GA, GD, NPO, RMU) differ in leakage and collateral damage across the popularity spectrum?

RQ3: Popularity vs. Forgettability. How much harder is it to erase popular facts than rare ones under identical unlearning settings?

4 Proposed UNLamb Benchmark

To systematically investigate the influence of knowledge popularity on machine unlearning, we introduce **UNLamb**, a new benchmark constructed from the PopQA dataset (Mallen et al., 2023). We chose PopQA as our foundation due to its direct mapping of natural language questions to structured Wikidata facts, which crucially includes a quantifiable popularity score for each entity. The QA format is particularly well-suited for unlearning evaluation: it allows precise measurement of whether a model can still retrieve a specific fact after unlearning, and is naturally sensitive to both failure modes, producing incoherent output under catastrophic forgetting and reproducing the correct answer verbatim under ineffective unlearning.

	1B	3B	8B
rare_forget15	0.010	0.027	0.051
popular_forget15	0.099	0.381	0.536

Table 1: ROUGE-L scores of original LLaMA-3 models of various sizes on 15% rare and popular forget-set splits.

UNLamb is derived from PopQA by restructuring its existing items into an unlearning-oriented benchmark: we retain the original questions, answers, and popularity scores without modification, and introduce a popularity-stratified partitioning into forget and retain splits suited for controlled unlearning experiments. This design choice ensures that the benchmark reflects organically acquired knowledge rather than artifacts of additional data collection or annotation. To further motivate the distinction between popular and rare knowledge, we construct the largest split of 15% rare and popular facts (which subsumes the 5% and 10% splits) and evaluate the original LLaMA-3 models of various sizes on this split (see Table 1); the results confirm that the models retain popular knowledge substantially better than rare knowledge.

This effect aligns with the broader observation that factual popularity correlates with entity frequency in large web corpora used for LLM pretraining. Sun et al. (Sun et al., 2023) show that models reliably memorize entities that appear often in sources like Wikipedia and DBpedia, while rarely mentioned entities are learned less robustly. In UNLamb, subject page views provide a direct signal of this exposure frequency, and Table 1 confirms that higher-popularity facts are recalled more consistently by the original models. This validates popularity as a meaningful proxy for memorization strength in our unlearning setting.

4.1 Corpus Definition and Popularity Metric

The UNLamb corpus $\mathcal{D}$ consists of $N = 11,600$ factual triplets, where each fact f_i is a tuple (q_i, s_i, p_i, a_i):

- $q_i \in \mathcal{Q}$ is a question in natural language (e.g., "What is the capital of France?").
- $s_i \in \mathcal{E}$ is the subject entity from Wikidata (e.g., "France").
- $p_i \in \mathcal{R}$ is the Wikidata property (e.g., `1082`, integer population).

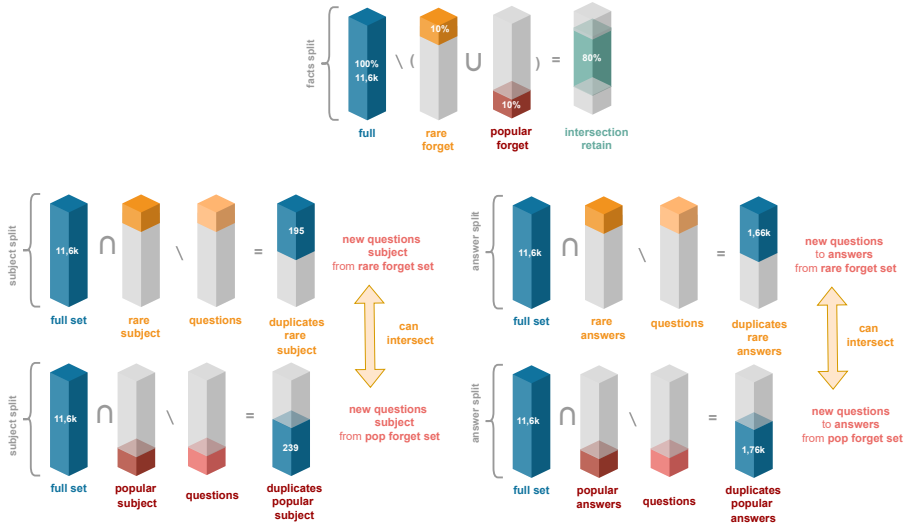

Figure 2: Benchmark construction pipeline for UNLamb with forget split of $p = 10\%$. **Top:** The full set of 11.6K facts is partitioned into two 10% "forget" splits (rare vs. popular) and an 80% "retain" split. **Left Bottom:** Subject-attack split: for each forget set, we select all facts from the full corpus sharing a subject entity with that forget set, excluding the forget set itself. **Right Bottom:** Answer-attack split: analogously for shared answers. The overlap between rare and popular attack splits is intentional, since the two forget sets are disjoint. Both splits measure collateral damage to knowledge related to the forgotten facts.

- $a_i \in \mathcal{A}$ is the single, canonical answer (e.g., "Paris").

For each fact f_i, its popularity score, $\text{pop}(f_i)$, is defined as the average monthly Wikipedia page views for its subject entity s_i. This metric serves as a strong proxy for a fact's prevalence, as Wikipedia is a core component of LLM pretraining corpora (Gao et al., 2020; Dodge et al., 2021). Our evaluation is therefore more realistic, focusing on the removal of organically acquired knowledge rather than on artifacts of fine-tuning.

By sorting the entire corpus $\mathcal{D}$ based on this score in ascending order, we obtain an ordered sequence of facts $f_{(1)}, f_{(2)}, \ldots, f_{(N)}$, where $\text{pop}(f_{(1)}) \leq \text{pop}(f_{(2)}) \leq \cdots \leq \text{pop}(f_{(N)})$.

4.2 Popularity-Stratified Forget and Retain Sets

The core of our benchmark design is to create unlearning tasks targeting the two extremes of the popularity spectrum. We select forgetting percentages $p \in \{5\%, 10\%, 15\%\}$. This choice is inspired by benchmarks like TOFU (Maini et al., 2024a), which uses splits of $\{1\%, 5\%, 10\%\}$, but we shift the range upwards because a 1% split provides too little data to produce stable, informative unlearning signals (Pal et al., 2025). For a given percentage p, we define the number of samples to forget from each extreme as $k = \lfloor \frac{p}{100} \cdot |\mathcal{D}| \rfloor$. Given our full benchmark size $|\mathcal{D}| = 11600$, this corresponds to forget sets (F_p^{rare}, F_p^{pop}) of 582 (5%), 1160 (10%), and 1750 (15%) samples, respectively. This allows us to define two distinct *forget sets*:

- **Rare Forget Set (F_p^{rare}):** The k least popular facts in the corpus.

$$F_p^{\text{rare}} = \{f_{(1)}, \ldots, f_{(k)}\} \tag{1}$$

- **Popular Forget Set (F_p^{pop}):** The k most popular facts in the corpus.

$$F_p^{\text{pop}} = \{f_{(D-k)}, \ldots, f_D\} \tag{2}$$

The remaining facts form the **Retain Set** R_p, which quantifies how much general knowledge remains preserved and untouched by the unlearning procedure.

$$R_p = \mathcal{D} \setminus \left(F_p^{\text{rare}} \cup F_p^{\text{pop}}\right) \tag{3}$$

4.3 Probing Unlearning Locality: Validation Splits

A successful unlearning algorithm should be precise, affecting only the targeted knowledge. To measure unintended side effects, or the *locality* of forgetting, we construct validation splits from the

retain set R_p. These splits contain facts that are related to the forget sets by either sharing the same subject or the same answer.

Let $S(F) = \{s_i \mid (q_i, s_i, p_i, a_i) \in F\}$ be the set of all subjects in a fact set F, and $A(F)$ be the set of all answers. We define the following validation sets:

- **Duplicate-Subject Sets (V_{DS}):** These sets test if the model forgets other facts about a subject $s(f)$ when one fact about it is unlearned.

$$V_{\text{DS},p}^{\text{rare}} = \{f \in R_p \mid s(f) \in S(F_p^{\text{rare}})\} \quad (4)$$
$$V_{\text{DS},p}^{\text{pop}} = \{f \in R_p \mid s(f) \in S(F_p^{\text{pop}})\} \quad (5)$$

- **Duplicate-Answer Sets (V_{DA}):** These sets test if recalling an answer is impaired when one question leading to it is unlearned.

$$V_{\text{DA},p}^{\text{rare}} = \{f \in R_p \mid a(f) \in A(F_p^{\text{rare}})\} \quad (6)$$
$$V_{\text{DA},p}^{\text{pop}} = \{f \in R_p \mid a(f) \in A(F_p^{\text{pop}})\} \quad (7)$$

Together, these splits measure both unlearning efficacy on target sets and collateral damage to related knowledge; Figure 2 illustrates the full construction for $p = 10\%$.

4.4 Evaluation Metrics

Unlearning Efficacy and Locality. We use **ROUGE-L Recall** (Lin, 2004; Geng et al., 2025) to measure word-level overlap between model output and the ground-truth answer. Lower scores on forget sets indicate more effective unlearning; higher (or FT-level) scores on retain and validation sets indicate better locality. As an additional semantic check, we report **Cosine Similarity (CS)** (Yuan et al., 2024) between pre- and post-unlearning outputs via Sentence-BERT (Reimers and Gurevych, 2019); lower CS reflects a larger semantic shift.

General Capability Preservation. We evaluate on MMLU (standard accuracy, `acc`) for broad world knowledge, and HellaSwag (normalized accuracy, `norm_acc`) for commonsense reasoning.

4.5 Unlearning Algorithms

Our benchmark includes a set of representative unlearning algorithms that address the unlearning objective by optimizing the model's parameters θ to remove the association between a question q and its specific answer a. For any given unlearning task, the target samples belong to a forget set $\mathcal{D}_f \in \{F_p^{\text{rare}}, F_p^{\text{pop}}\}$.

Gradient Ascent (GA) (Jang et al., 2022). A foundational technique that reverses the learning process by maximizing the Negative Log-Likelihood (NLL) on the forget set. This is equivalent to performing gradient descent on the negative of the standard training loss. We define the NLL loss over a benchmark $\mathcal{D}$ as

$$\mathcal{L}_{\text{NLL}}(\mathcal{D}, \theta) = -\frac{1}{|\mathcal{D}|} \sum_{(q,a)\in\mathcal{D}} \log P_\theta(a|q). \quad (8)$$

The final loss for GA is the negative of this loss on the forget set:

$$\mathcal{L}^{\text{GA}}(\theta) = -\mathcal{L}_{\text{NLL}}(\mathcal{D}_f, \theta) \quad (9)$$

Gradient Difference (GD) (Liu et al., 2022) extends GA by adding a regularization term to maintain performance on the retain set, $\mathcal{D}_r$. It simultaneously maximizes the loss on the forget data while minimizing it on the retain data, anchoring the model to its useful knowledge. The loss function is:

$$\mathcal{L}^{\text{GD}}(\theta) = -\mathcal{L}_{\text{NLL}}(\mathcal{D}_f, \theta) + \lambda \cdot \mathcal{L}_{\text{NLL}}(\mathcal{D}_r, \theta) \quad (10)$$

Negative Preference Optimization (NPO) (Zhang et al., 2024) treats unlearning as a preference task. The objective is to make the probability of the undesired answer, $P_\theta(a|q)$, as small as possible relative to a reference model $P_{\theta_{\text{ref}}}(a|q)$. The loss function is formulated as:

$$\mathcal{L}^{\text{NPO}}(\theta) = \frac{2}{\beta} E_{(q,a)\in\mathcal{D}_f}\left[\log\left(1 + \left(\frac{P_\theta(a|q)}{P_{\theta_{\text{ref}}}(a|q)}\right)^\beta\right)\right], \quad (11)$$

where β is a hyperparameter we set to 1, as suggested in the original work.

Representation Misdirection for Unlearning (RMU) (Li et al., 2024) operates on the model's internal hidden representations rather than its output probabilities. The core idea is to push the hidden states of the forget data towards a misdirection vector while ensuring the hidden states for the retain data remain close to those of the original model. The total loss is a weighted sum of a forget loss $\mathcal{L}_F$ and a retain loss $\mathcal{L}_R$.

$$\mathcal{L}_F = E_{(q,a)\in\mathcal{D}_f}\left[\frac{1}{|a|}\sum_{t\in a} \|h_\theta(t) - c \cdot u\|_2^2\right] \quad (12)$$

$$\mathcal{L}_R = E_{(q,a)\in\mathcal{D}_r}\left[\frac{1}{|a|}\sum_{t\in a} \|h_\theta(t) - h_{\theta_{\text{ref}}}(t)\|_2^2\right] \quad (13)$$

$$\mathcal{L}^{\text{RMU}}(\theta) = \mathcal{L}_F + \lambda \cdot \mathcal{L}_R \quad (14)$$

Here, $h_\theta(t)$ is the hidden state for token t of the answer in the current model, $h_{\theta_{\text{ref}}}(t)$ is the state in

the original reference model, u is typically a predefined random vector for misdirection, and c is a scaling factor.

Earlier MU methods, such as In-Context Unlearning (ICU) (Pawelczyk et al., 2024) and Rejection Tuning (RT) (Maini et al., 2024b), have become outdated in practice and are superseded by optimization-based approaches like NPO; we therefore keep GA and GD as canonical baselines and include NPO and RMU, while treating ICU and RT as deprecated.

5 Experiments

We evaluate four unlearning algorithms (GA, GD, NPO, RMU) on LLaMA-3 models (1B, 3B, 8B) across rare and popular forget splits of 5%, 10%, and 15%. All models are first fine-tuned on the full UNLamb corpus before unlearning begins, ensuring a consistent and knowledge-rich starting point.

5.1 Experimental Details

Compute budget and evaluation cost. One full experiment cycle on an A100 GPU (comprising $\approx$ 1h10m for fine-tuning, $\approx$ 1.5h for unlearning with GA, GD, NPO, and RMU, and $\approx$ 9h for ROUGE evaluation) takes approximately 12 hours.

For additional semantic validation, we also report BERTScore (Zhang et al., 2020) for the best LLaMA and Gemma configurations (see Table 3). According to BERTScore, the degradation in answer quality is notably stronger when forgetting popular facts, reinforcing the same trend observed with ROUGE. We intentionally exclude LLM-based judgments because they are computationally expensive and their results vary across providers and over time, making them unsuitable for reproducible large-scale evaluation.

Model Preparation. Our base model for all unlearning experiments is a LLaMA-3 model fine-tuned on the complete UNLamb corpus $\mathcal{D}$, creating a consistent, knowledge-rich starting point. Fine-tuning was performed for 5 epochs with a learning rate of 2×10^{-4} using Low-Rank Adaptation (LoRA) with a rank of 32 and an alpha of 64. To ensure reproducibility, all experiments use a fixed random seed (42).

We additionally evaluate Gemma 7B on the 10% split (Table 2): it achieves effective unlearning for rare facts but fails on popular ones under identical settings, likely due to differences in architecture or pretraining.

	FT	GA	GD	NPO	RMU
Rare unlearning					
Forget ↓	0.46	**0.25**	**0.32**	**0.32**	**0.32**
Dup. subjects ↑	0.43	0.33	0.42	0.43	0.31
Dup. answers ↑	0.73	0.57	0.65	0.65	0.57
Retain ↑	0.50	**0.40**	**0.47**	**0.46**	**0.38**
Popular unlearning					
Forget ↓	0.91	**0.85**	**0.89**	**0.87**	**0.74**
Dup. subjects ↑	0.47	0.47	0.49	0.47	0.20
Dup. answers ↑	0.74	0.73	0.74	0.74	0.58
Retain ↑	0.50	**0.50**	**0.51**	**0.51**	**0.32**

Table 2: ROUGE-L recall on Gemma 7B for 10% forget-set size. Top: rare unlearning splits; Bottom: popular unlearning splits.

	FT		GA		GD		NPO		RMU	
	L8B	G7B	L8B	G7B	L8B	G7B	L8B	G7B	L8B	G7B
Rare unlearning										
Forget ↓	0.41	0.36	**0.31**	**0.27**	**0.40**	**0.30**	**0.36**	**0.30**	**0.41**	**0.28**
Dup. subjects ↑	0.44	0.36	0.30	0.36	0.45	0.36	0.40	0.37	0.44	0.28
Dup. answers ↑	0.41	0.45	0.35	0.40	0.41	0.43	0.40	0.43	0.41	0.38
Retain ↑	0.43	0.39	**0.39**	**0.35**	**0.43**	**0.38**	**0.42**	**0.37**	**0.44**	**0.32**
Popular unlearning										
Forget ↓	0.42	0.58	**0.16**	**0.55**	**0.41**	**0.56**	**0.23**	**0.56**	**0.41**	**0.55**
Dup. subjects ↑	0.43	0.36	0.29	0.36	0.42	0.38	0.37	0.36	0.43	0.30
Dup. answers ↑	0.40	0.47	0.26	0.47	0.41	0.47	0.34	0.47	0.41	0.40
Retain ↑	0.43	0.39	**0.23**	**0.39**	**0.43**	**0.39**	**0.37**	**0.39**	**0.43**	**0.34**

Table 3: BERTScore on UNLamb for 10% forget-set. Top: rare unlearning splits; Bottom: popular unlearning splits. L8B = LLaMA-3.1 8B, G7B = Gemma 7B.

5.2 Hyperparameter Optimization

We ran a grid search over epochs (1–5), LoRA rank and alpha (4–256), and learning rate (10^{-5}–5×10^{-3}), yielding an optimal fine-tuning rate of $\mathbf{2 \times 10^{-4}}$, an unlearning rate of $\mathbf{3 \times 10^{-5}}$, and LoRA rank 32 with $\alpha = 64$. Full configuration files are provided in the repository.

Unlearning Procedure. Starting from the fine-tuned model, we perform unlearning with LoRA (rank=32, $\alpha = 64$), selected via grid search on the rare and popular forget sets. Each unlearning run uses the grid-searched configuration and is carried out for two epochs: the search revealed that one epoch is typically insufficient to induce effective forgetting, whereas three or more cause catastrophic degradation of model utility. After unlearning on each set, we compute the full suite of metrics (forget-set efficacy, validation-locality, and general capability preservation) and analyze the differences between rare and popular fact removal.

5.3 Results

Unlearning Utility by Forget Set Size. We evaluated unlearning with forget-set sizes of 5% (582 samples), 10% (1.16k), and 15% (1.75k); results in Table 4 report efficacy and retention across methods. The 10% split yielded the best trade-off for both rare and popular facts: 5% was too small to

	FT			GradAscent			GradDiff			NPO			RMU		
Split, p%	5%	10%	15%	5%	10%	15%	5%	10%	15%	5%	10%	15%	5%	10%	15%
Rare unlearning															
rare_forget_p% ↓	0.967	0.969	0.973	**0.905**	**0.633**	**0.000**	**0.915**	**0.852**	**0.454**	**0.905**	**0.763**	**0.047**	**0.781**	**0.726**	**0.741**
duplicate_subjects_rare_forget_p% ↑	0.978	0.982	0.990	0.989	0.836	0.000	0.989	0.973	0.770	0.989	0.952	0.197	0.757	0.518	0.272
duplicate_answers_rare_forget_p% ↑	0.994	0.988	0.993	0.993	0.765	0.000	0.994	0.979	0.720	0.993	0.917	0.176	0.736	0.597	0.544
retain_intersection_(100-2p)% ↑	0.989	0.989	0.990	**0.988**	**0.867**	**0.000**	**0.988**	**0.978**	**0.728**	**0.988**	**0.936**	**0.187**	**0.777**	**0.622**	**0.354**
Popular unlearning															
popular_forget_p% ↓	1.000	1.000	0.999	**0.993**	**0.191**	**0.000**	**1.000**	**0.998**	**0.183**	**0.993**	**0.411**	**0.012**	**0.962**	**0.882**	**0.818**
duplicate_subjects_popular_forget_p% ↑	1.000	1.000	0.998	0.990	0.318	0.000	0.990	1.000	0.429	1.000	0.773	0.116	0.810	0.389	0.176
duplicate_answers_popular_forget_p% ↑	0.997	0.989	0.990	0.996	0.564	0.000	0.995	0.988	0.597	0.994	0.834	0.231	0.950	0.696	0.493
retain_intersection_(100-2p)% ↑	0.989	0.989	0.990	**0.988**	**0.409**	**0.000**	**0.989**	**0.990**	**0.546**	**0.989**	**0.831**	**0.139**	**0.890**	**0.596**	**0.310**

Table 4: Unlearning LLaMA-3.1 8B: ROUGE-L Recall of five methods (FT, GradAscent, GradDiff, NPO, RMU) for 5–15% forget-set sizes, on rare (top) vs. popular (bottom) splits. Learning rate: 3×10^{-5}.

	FT			GradAscent			GradDiff			NPO			RMU		
Model size, B	1B	3B	8B	1B	3B	8B	1B	3B	8B	1B	3B	8B	1B	3B	8B
Rare unlearning															
rare_forget10 ↓	0.160	0.211	0.969	**0.019**	**0.051**	**0.633**	**0.130**	**0.048**	**0.852**	**0.074**	**0.047**	**0.763**	**0.128**	**0.186**	**0.726**
duplicate_subjects_rare_forget10 ↑	0.132	0.283	0.982	0.069	0.165	0.836	0.094	0.246	0.973	0.091	0.197	0.952	0.070	0.202	0.518
duplicate_answers_rare_forget10 ↑	0.353	0.518	0.988	0.175	0.279	0.765	0.356	0.483	0.979	0.269	0.400	0.917	0.295	0.363	0.597
retain_intersection80 ↑	0.129	0.341	0.989	**0.056**	**0.133**	**0.867**	**0.128**	**0.322**	**0.978**	**0.091**	**0.228**	**0.936**	**0.087**	**0.236**	**0.622**
Popular unlearning															
popular_forget10 ↓	0.456	0.816	1.000	**0.100**	**0.087**	**0.191**	**0.325**	**0.525**	**0.998**	**0.129**	**0.535**	**0.411**	**0.354**	**0.532**	**0.882**
duplicate_subjects_popular_forget10 ↑	0.115	0.337	1.000	0.056	0.036	0.318	0.115	0.258	1.000	0.066	0.055	0.773	0.095	0.192	0.389
duplicate_answers_popular_forget10 ↑	0.348	0.556	0.989	0.272	0.246	0.564	0.343	0.530	0.988	0.291	0.308	0.834	0.312	0.468	0.696
retain_intersection80 ↑	0.129	0.341	0.989	**0.077**	**0.088**	**0.409**	**0.122**	**0.322**	**0.990**	**0.090**	**0.127**	**0.831**	**0.101**	**0.247**	**0.596**

Table 5: Unlearning LLaMA-3 models (1B–8B): ROUGE-L Recall of five methods (FT, GradAscent, GradDiff, NPO, RMU) on 10% rare (top) vs. popular (bottom) forget sets.

reliably induce forgetting, whereas 15% frequently triggered catastrophic degradation or ineffective unlearning unless hyperparameters were carefully retuned (see detailed hyperparameter behavior in Table 7). Crucially, catastrophic forgetting at 15% was considerably more severe when removing popular facts, suggesting that the optimal forget-set size itself depends on fact popularity: a setting safe for rare facts may be too aggressive for popular ones.

Hyperparameter Sensitivity. Both learning rate and the number of training epochs control the overall strength of the unlearning signal: higher values of either increase forgetting efficacy but raise the risk of catastrophic collapse. Epochs are the coarser and more aggressive lever, as each additional epoch accumulates gradient updates globally and is more likely to trigger model-wide degradation. Learning rate provides finer-grained control over the per-step update magnitude and is therefore a safer axis for tuning unlearning intensity. LoRA rank and α had comparatively minor influence once set to rank 32, α=64. Critically, popular facts consistently require more aggressive settings to achieve the same forget-set reduction as rare facts, narrowing the safe hyperparameter window and making per-popularity tuning essential.

Unlearning Utility by Model Size. We analyzed how model size impacts unlearning efficacy (Table 5). We compared four unlearning methods (GA, GD, NPO, RMU) on LLaMA-3 models with 1B, 3B, and 8B parameters, using a 10% forget split, which is a standard practice in unlearning studies (Maini et al., 2024b; Pal et al., 2025).

Unlearning effectiveness depends strongly on model size. Smaller models (1B and 3B) behave inconsistently: GA produces extreme forgetting with severe collateral damage to the retain set, while GD and NPO achieve only modest reductions in forget-set ROUGE-L. The 8B model encodes facts more strongly and exhibits a clearer popularity gap. When targeting popular facts, it is far more prone to catastrophic forgetting, especially with Gradient Ascent, than when targeting rare ones. This pattern implies that stronger memorization leads to more brittle unlearning: the 8B model's deeper encoding of popular knowledge is precisely what makes its removal more disruptive. Taken together, these results suggest that both model scale and fact popularity must be considered jointly when selecting an unlearning strategy.

Unlearning LLMs on Benchmarks. Despite drops in UNLamb-specific metrics, overall model quality remains largely intact (Table 6). For LLaMA-3 8B at 10%, MMLU scores decline by at most 1.1 points (FT 0.670 → 0.659 for GA

	FT			GradAscent			GradDiff			NPO			RMU		
	1B	3B	8B	1B	3B	8B	1B	3B	8B	1B	3B	8B	1B	3B	8B
Rare unlearning															
MMLU ↑	0.435	0.601	0.670	0.389	0.593	0.670	0.385	0.588	0.668	0.388	0.593	0.669	0.349	0.579	0.655
HellaSwag ↑	0.614	0.711	0.788	0.578	0.658	0.757	0.581	0.662	0.747	0.583	0.660	0.753	0.581	0.668	0.750
Popular unlearning															
MMLU ↑	0.435	0.601	0.670	0.386	0.579	0.659	0.390	0.585	0.666	0.387	0.581	0.662	0.352	0.581	0.649
HellaSwag ↑	0.614	0.711	0.788	0.581	0.668	0.749	0.580	0.666	0.744	0.581	0.668	0.749	0.580	0.670	0.752

Table 6: MMLU and HellaSwag scores after 10% data unlearning across rare vs. popular splits and three LLaMA-3 model sizes.

	FT	GradAscent			GradDiff			NPO			RMU		
Unlearning lr	–	1e-5	2e-5	3e-5	1e-5	2e-5	3e-5	1e-5	2e-5	3e-5	1e-5	2e-5	3e-5
Rare unlearning													
rare_forget15 ↓	0.973	0.941	0.021	0.000	0.959	0.569	**0.454**	0.954	0.316	0.047	0.931	0.895	0.741
retain_intersection70 ↑	0.990	0.985	0.038	0.000	0.988	0.772	**0.728**	0.988	0.492	0.187	0.978	0.909	0.354
Popular unlearning													
popular_forget15 ↓	0.999	**0.499**	0.012	0.000	0.997	0.209	0.183	0.942	0.059	0.012	0.986	0.979	0.818
retain_intersection70 ↑	0.990	**0.842**	0.019	0.000	0.990	0.562	0.546	0.985	0.223	0.139	0.974	0.943	0.310

Table 7: ROUGE-L recall for 15% rare vs. popular facts unlearning with varying unlearning learning rates.

on popular facts) and HellaSwag by at most 3.9 points (0.788 → 0.749). Crucially, even GA, which causes catastrophic UNLamb forgetting, leaves downstream benchmarks largely intact, suggesting that unlearning-induced forgetting is relatively localized to the target domain.

6 Discussion

Below, we summarize the answers to the three research questions raised in Section 3.1 and highlight practical conclusions.

RQ1: Scale Effects. The 1B and 3B models memorize rare facts poorly (FT ROUGE-L 0.160 and 0.211), limiting the interpretability of unlearning at those scales. The 8B model shows strong memorization of both fact types (0.969 and 1.000), and is the only size at which the popularity gap is clearly observable. At 10%, GA achieves forget 0.633 / retain 0.867 on rare facts, but collapses retain to 0.409 on popular ones; at 15%, it causes complete model collapse, especially for popular facts. The 8B scale is thus the minimum meaningful testbed for popularity-aware unlearning studies.

Takeaway 1: Machine unlearning is most effective on larger models (at least 8B). **A forget set of ∼10% yields optimal and more stable results**; smaller sets are insufficient, while larger ones tend to cause unstable unlearning and require more careful parameter tuning.

RQ2: Algorithmic Robustness. At 10%, GA best reduces rare-fact ROUGE-L (0.633) but collapses retain on popular facts (0.409); NPO offers the best popular-fact balance (forget 0.411, retain 0.831); GD barely forgets popular facts at all (0.998). At 15%, GD is preferred for rare (forget 0.454, retain 0.728) and GA for popular (forget 0.499, retain 0.842). The learning rate is the dominant hyperparameter: a single step from 1×10^{-5} to 2×10^{-5} flips GA at 15% from ineffective (forget 0.941) to catastrophic (forget 0.021, retain 0.038). No algorithm can thus be safely deployed without per-popularity hyperparameter validation.

Takeaway 2: Effective unlearning depends on fact popularity and forget-set size: for 10%, use GradAscent on rare and NPO on popular; for 15%, use GradDiff on rare and GradAscent on popular. **Learning rate is the key parameter**, most affecting unlearning, and **two epochs strike the best balance** between forgetting and stability.

RQ3: Popularity vs. Forgettability. LLaMA-3 8B scores ROUGE-L 0.536 on the popular 15% split versus 0.051 on the rare split (Table 1), a tenfold memorization gap rooted in pretraining frequency. This gap propagates into unlearning: popular-fact removal consistently incurs greater collateral damage (duplicate-subject retention 0.773 vs. 0.952 for rare), and RMU, which achieves forget 0.726 on rare facts, barely moves popular ones (0.882). Popular facts resist localized gradient interventions precisely because they are encoded redundantly across pretraining.

Cross-Architecture Generalization. The Gemma 7B results (Table 2) corroborate the LLaMA findings. Under identical 10% settings, Gemma achieves partial rare-fact unlearning (GA reduces forget ROUGE-L from 0.458 to 0.246) while all algorithms largely fail on popular facts (GA: 0.854, GD: 0.885, NPO: 0.868 from FT 0.910). This mirrors LLaMA-3 8B behavior and suggests that the popularity barrier generalizes across model families, rather than being an artifact of a specific architecture, pointing to a fundamental property of how pretraining exposure frequency shapes the resistance of knowledge to gradient-based removal.

Takeaway 3: Fact popularity strongly affects unlearning difficulty. **Rare facts are easier to forget cleanly; popular facts tend toward catastrophic forgetting** unless handled carefully.

Failure Mode Analysis. We observe two qualitatively distinct failure modes across all algorithms. *Catastrophic forgetting* occurs when unlearning is overly aggressive: the model loses coherent generation ability on both forget and retain sets, producing random tokens regardless of the input. This is most pronounced with Gradient Ascent at high learning rates or large forget-set sizes (Table 7), and is consistent with the known instability of unconstrained gradient ascent on deeply memorized knowledge. *Ineffective unlearning*, by contrast, leaves the targeted fact easily recoverable while general capabilities remain intact; this arises when the unlearning signal is insufficient relative to the memorization strength of the fact. Popular facts are disproportionately prone to both failure modes: their redundant encoding across pretraining documents demands a stronger unlearning signal to suppress, increasing the risk of catastrophic collapse, while weaker signals leave the fact easily recoverable, narrowing the safe operating window and making per-popularity hyperparameter tuning essential.

7 Conclusion

In this work, we construct UNLamb, a QA-based benchmark for studying how fact popularity affects unlearning in LLMs. Across four methods, three forget-set sizes in both rare and popular splits, and three model sizes, we find that popular facts are harder to remove and more likely to cause catastrophic forgetting, while rare facts are erased more cleanly. These results reveal a systematic gap in current methods and lay the foundation for adaptive schemes, such as dynamic learning rate selection and popularity-aware method selection, to improve unlearning precision and stability. We hope UNLamb will serve as a standard testbed for evaluating popularity-aware unlearning strategies and encourage the community to account for knowledge prevalence when designing and benchmarking future unlearning algorithms. Future work should explore adaptive MU strategies for integrating popularity information into unlearning algorithms.

8 Limitations

It remains unclear how far the results generalize across model families, sizes, and MoE architectures; extending the evaluation to these settings is an open direction for future work. A detailed error analysis of facts that resist parametric unlearning is also absent. Future work should also align the studied unlearning methods with theoretical work on data distributions and memorization (Yu et al., 2024; Sander et al.; Mireshghallah and Li, 2025; Fan et al., 2025). Additionally, while MMLU and HellaSwag results (Tables 6, 8) suggest that downstream reasoning capabilities remain largely intact after unlearning, investigating propagation to open-ended dialogue and other generative tasks remains an open direction.

Ethics Statement

All data are sourced from public repositories; no sensitive attributes were collected, and no human subjects were involved. We regard machine unlearning as a contribution to AI safety and data governance, as it enables the removal of unsafe, outdated, or private content from deployed models. ChatGPT was used solely for minor language and grammatical editing; all research design, analysis, and interpretation were performed by the authors.

Acknowledgments

The work was supported by a grant, provided by the Ministry of Economic Development of the Russian Federation in accordance with the subsidy agreement (agreement identifier 000000C313925P4G0002) and the agreement with the Ivannikov Institute for System Programming of the Russian Academy of Sciences dated June 20, 2025 No. 139-15-2025-011.

References

Yinzhi Cao and Junfeng Yang. 2015. Towards making systems forget with machine unlearning. In *2015 IEEE symposium on security and privacy*, pages 463–480. IEEE.

Roi Cohen, Eden Biran, Ori Yoran, Amir Globerson, and Mor Geva. 2024. Evaluating the ripple effects of knowledge editing in language models. *Transactions of the Association for Computational Linguistics*, 12:283–298.

Jesse Dodge, Maarten Sap, Ana Marasović, William Agnew, Gabriel Ilharco, Dirk Groeneveld, Margaret Mitchell, and Matt Gardner. 2021. Documenting large webtext corpora: A case study on the colossal clean crawled corpus. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Bowen Fan, Yuming Ai, Xunkai Li, Zhilin Guo, Rong-Hua Li, and Guoren Wang. 2025. Opengu: A comprehensive benchmark for graph unlearning. *arXiv preprint arXiv:2501.02728*.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, and 1 others. 2020. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*.

Jiahui Geng, Qing Li, Herbert Woisetschlaeger, Zongxiong Chen, Fengyu Cai, Yuxia Wang, Preslav Nakov, Hans-Arno Jacobsen, and Fakhri Karray. 2025. A comprehensive survey of machine unlearning techniques for large language models. *arXiv preprint arXiv:2503.01854*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Shengyuan Hu, Yiwei Fu, Zhiwei Steven Wu, and Virginia Smith. 2024. Unlearning or obfuscating? jogging the memory of unlearned llms via benign relearning. *arXiv preprint arXiv:2406.13356*.

Joel Jang, Dongkeun Yoon, Sohee Yang, Sungmin Cha, Moontae Lee, Lajanugen Logeswaran, and Minjoon Seo. 2022. Knowledge unlearning for mitigating privacy risks in language models. *arXiv preprint arXiv:2210.01504*.

Joel Jang, Dongkeun Yoon, Sohee Yang, Sungmin Cha, Moontae Lee, Lajanugen Logeswaran, and Minjoon Seo. 2023. Knowledge unlearning for mitigating privacy risks in language models. In *ACL 2023*, pages 14389–14408. Association for Computational Linguistics (ACL).

Zhuoran Jin, Pengfei Cao, Chenhao Wang, Zhitao He, Hongbang Yuan, Jiachun Li, Yubo Chen, Kang Liu, and Jun Zhao. 2024. Rwku: Benchmarking real-world knowledge unlearning for large language models. In *Advances in Neural Information Processing Systems*, volume 37, pages 98213–98263. Curran Associates, Inc.

Meghdad Kurmanji, Peter Triantafillou, Jamie Hayes, and Eleni Triantafillou. 2023. Towards unbounded machine unlearning. *Advances in neural information processing systems*, 36:1957–1987.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D Li, Ann-Kathrin Dombrowski, Shashwat Goel, Long Phan, and 1 others. 2024. The wmdp benchmark: Measuring and reducing malicious use with unlearning. *arXiv preprint arXiv:2403.03218*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Bo Liu, Qiang Liu, and Peter Stone. 2022. Continual learning and private unlearning. In *Conference on Lifelong Learning Agents*, pages 243–254. PMLR.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C Lipton, and J Zico Kolter. 2024a. Tofu: A task of fictitious unlearning for llms. *arXiv preprint arXiv:2401.06121*.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C. Lipton, and J. Zico Kolter. 2024b. Tofu: A task of fictitious unlearning for llms.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9802–9822.

Alessandro Mantelero. 2013. The eu proposal for a general data protection regulation and the roots of the â right to be forgottenâ. *Computer Law & Security Review*, 29(3):229–235.

Niloofar Mireshghallah and Tianshi Li. 2025. Position: Privacy is not just memorization! *arXiv preprint arXiv:2510.01645*.

Soumyadeep Pal, Changsheng Wang, James Diffenderfer, Bhavya Kailkhura, and Sijia Liu. 2025. Llm unlearning reveals a stronger-than-expected coreset effect in current benchmarks. *arXiv e-prints*, pages arXiv–2504.

Martin Pawelczyk, Seth Neel, and Himabindu Lakkaraju. 2024. In-context unlearning: Language models as few-shot unlearners. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 40034–40050. PMLR.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, page 3982. Association for Computational Linguistics.

Jie Ren, Zhenwei Dai, Xianfeng Tang, Yue Xing, Shenglai Zeng, Hui Liu, Jingying Zeng, Qiankun Peng, Samarth Varshney, Suhang Wang, and 1 others. 2025. Keeping an eye on llm unlearning: The hidden risk and remedy. *arXiv preprint arXiv:2506.00359*.

Tom Sander, Bargav Jayaraman, Mark Ibrahim, Kamalika Chaudhuri, and Chuan Guo. Rethinking the role of verbatim memorization in llm privacy. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Ayush Sekhari, Jayadev Acharya, Gautam Kamath, and Ananda Theertha Suresh. 2021. Remember what you want to forget: Algorithms for machine unlearning. *Advances in Neural Information Processing Systems*, 34:18075–18086.

Thanveer Shaik, Xiaohui Tao, Haoran Xie, Lin Li, Xiaofeng Zhu, and Qing Li. 2024. Exploring the landscape of machine unlearning: A comprehensive survey and taxonomy. *IEEE Transactions on Neural Networks and Learning Systems*, 36(7):11676–11696.

Weijia Shi, Jaechan Lee, Yangsibo Huang, Sadhika Malladi, Jieyu Zhao, Ari Holtzman, Daogao Liu, Luke Zettlemoyer, Noah A Smith, and Chiyuan Zhang. 2024. Muse: Machine unlearning six-way evaluation for language models. *CoRR*.

Nianwen Si, Hao Zhang, Heyu Chang, Wenlin Zhang, Dan Qu, and Weiqiang Zhang. 2023. Knowledge unlearning for llms: Tasks, methods, and challenges. *arXiv preprint arXiv:2311.15766*.

Kai Sun, Yifan Ethan Xu, Hanwen Zha, Yue Liu, and Xin Luna Dong. 2023. Head-to-tail: How knowledgeable are large language models (llm). *AKA will llms replace knowledge graphs*.

Lingzhi Wang, Xingshan Zeng, Jinsong Guo, Kam-Fai Wong, and Georg Gottlob. 2025. Selective forgetting: Advancing machine unlearning techniques and evaluation in language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 843–851.

Boyi Wei, Weijia Shi, Yangsibo Huang, Noah A Smith, Chiyuan Zhang, Luke Zettlemoyer, Kai Li, and Peter Henderson. 2024. Evaluating copyright takedown methods for language models. *Advances in Neural Information Processing Systems*, 37:139114–139150.

Lijia Yu, Xiao-Shan Gao, Lijun Zhang, and Yibo Miao. 2024. Generalizablity of memorization neural network. *Advances in Neural Information Processing Systems*, 37:113311–113359.

Hongbang Yuan, Zhuoran Jin, Pengfei Cao, Yubo Chen, Kang Liu, and Jun Zhao. 2025. Towards robust knowledge unlearning: An adversarial framework for assessing and improving unlearning robustness in large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 25769–25777.

Xiaojian Yuan, Tianyu Pang, Chao Du, Kejiang Chen, Weiming Zhang, and Min Lin. 2024. A closer look at machine unlearning for large language models. *arXiv preprint arXiv:2410.08109*.

David Zagardo. 2024. A more practical approach to machine unlearning. *arXiv preprint arXiv:2406.09391*.

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. 2024. Negative preference optimization: From catastrophic collapse to effective unlearning. *arXiv preprint arXiv:2404.05868*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Kairan Zhao, Meghdad Kurmanji, George-Octavian Barbulescu, Eleni Triantafillou, and Peter Triantafillou. 2024. What makes unlearning hard and what to do about it. In *Advances in Neural Information Processing Systems*, volume 37, pages 12293–12333. Curran Associates, Inc.

A Additional Results

Performance on downstream LLM benchmarks (MMLU, HellaSwag) remains largely unaffected following both 5% and 15% unlearning (see Table 8). The sole exception is Gradient Ascent at the 15% level, which exhibits an approximate 30% reduction in performance, consistent with catastrophic forgetting.

Table 8: LLM performance after 5% and 15% data unlearning: MMLU and HellaSwag on rare vs. popular splits (LLaMA-3.1 8B).

	GradAscent		**GradDiff**		**NPO**		**RMU**	
	5%	**15%**	**5%**	**15%**	**5%**	**15%**	**5%**	**15%**
Rare unlearning								
MMLU ↑	0.548	0.382	0.548	0.551	0.547	0.551	0.533	0.519
HellaSwag ↑	0.663	0.518	0.663	0.661	0.663	0.658	0.670	0.670
Popular unlearning								
MMLU ↑	0.547	0.381	0.548	0.538	0.548	0.540	0.523	0.528
HellaSwag ↑	0.663	0.603	0.662	0.662	0.662	0.661	0.667	0.668

We study how varying the regularization weight λ in Gradient Difference impacts unlearning on the 10% forget split and observe a pronounced divergence between popular and rare facts (see Table 9). Popular facts exhibit a sharp trade-off: small λ values enable forgetting the target set but induce

catastrophic forgetting, whereas larger λ values preserve retention at the cost of ineffective unlearning. In contrast, rare facts behave smoothly (there is no abrupt collapse), and tuning λ yields a gradual trade-off between forgetting and retention. These results suggest that unlearning popular facts is substantially more difficult and demands careful, potentially adaptive, selection of λ, whereas rare facts are comparatively robust to their setting.

Table 9: GradDiff sensitivity to the regularization weight λ on the 10% forget and 80% retain splits (LLaMA-3.1 8B).

	Forget 10% (ROUGE-L ↓)		**Retain 80% (ROUGE-L ↑)**	
λ	**Rare**	**Popular**	**Rare**	**Popular**
0.10	0.71	0.23	0.90	0.54
0.20	0.74	0.38	0.92	0.78
0.30	0.77	0.68	0.94	0.94
0.40	0.80	0.92	0.96	0.99
0.60	0.82	0.98	0.97	1.00