UNLEARNING THAT LASTS: UTILITY-PRESERVING, ROBUST, AND ALMOST IRREVERSIBLE FORGETTING IN LLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Unlearning in large language models (LLMs) involves precisely removing specific information from a pre-trained model. This is crucial to ensure safety of LLMs by deleting private data or harmful knowledge acquired during pre-training. However, existing unlearning methods often fall short when subjected to thorough evaluations. To overcome this, we introduce JensUn, where we leverage the Jensen-Shannon Divergence as the training objective for both forget and retain sets for more stable and effective unlearning dynamics compared to commonly used loss functions. In extensive experiments, JensUn achieves better forget-utility trade-off than competing methods, and even demonstrates strong resilience to benign relearning. Additionally, for a precise unlearning evaluation, we introduce LKF, a curated dataset of lesser-known facts that provides a realistic unlearning scenario. Finally, to comprehensively test unlearning methods, we propose *(i)* employing an LLM as semantic judge instead of the standard ROUGE score, and *(ii)* using worst-case unlearning evaluation over various paraphrases and input formats. Our improved evaluation framework reveals that many existing methods are less effective than previously thought.

1 INTRODUCTION

Training large language models (LLMs) on massive data scraped from the internet yields impressive performance but comes with serious safety concerns, including the risk of exposing private information (Nasr et al., 2023), violating copyrights (Wu et al., 2023; Jang et al., 2023; Karamolegkou et al., 2023), and amplifying harmful content (Huang et al., 2024; Lu et al., 2022; Barrett et al., 2023; Wen et al., 2023). To prevent acquisition of undesired knowledge, one could selectively remove or adjust problematic samples in the training data and then re-train LLMs from scratch. Since this is an expensive process, recent works have explored more efficient alternatives, such as model editing and machine unlearning. In contrast to re-training, these approaches aim to update a pre-trained LLM to remove or change the internal knowledge encoded in its parameters. While model editing is used to update the model for a specific piece of existing information (Meng et al., 2022; Ilharco et al., 2023), *machine unlearning* aims to remove entire concepts from the model (Liu et al., 2025), like dangerous information (Li et al., 2024; Barrett et al., 2023), and private sensitive data (Nasr et al., 2023), or tries to make the model adhere to the right to be forgotten (Zhang et al., 2024a). Given its practical relevance in these high-stakes scenarios, many approaches to machine unlearning have appeared (Jang et al., 2023; Rafailov et al., 2023; Fan et al., 2024; Li et al., 2024). However, evaluating their effectiveness is a delicate task, since it has to be determined if the relevant information has been truly forgotten, or if the model simply suppresses it at a superficial level without actually removing it (Hu et al., 2024; Thaker et al., 2025) and it can be easily re-introduced by fine-tuning on new data (Hu et al., 2024).

In this work, we propose a *new unlearning method based on Jensen-Shannon Divergence*, termed **JensUn**. LLMs unlearned with JensUn demonstrate better forget-utility trade-off than the state-of-the-art baselines (see left plot in Figure 1). In fact, our models attain the best unlearning quality (under our proposed strong worst-case evaluation) while preserving the highest utility on average across different utility metrics, LLMs, and unlearning datasets. Moreover, JensUn yields the highest robustness to *benign relearning* (Lucki et al., 2024; Hu et al., 2024). That is, the LLMs do not recover

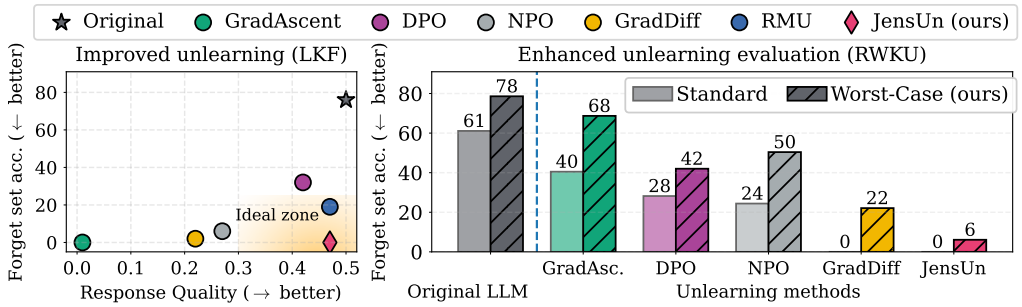

Figure 1: **Our JensUn yields the best trade-off between unlearning quality (forget set accuracy) and utility of the LLM.** *(left)* Our unlearning method JensUn achieves on our LKF dataset an optimal worst-case forget set accuracy of 0% while maintaining high response quality (AlpacaEval), the most similar to the original Llama-3.2-3B-Instruct pre-trained model. *(right)* Our novel worst-case evaluation using 15 paraphrases of the query on RWKU reveals that using single question-answer evaluations overestimates unlearning quality: our worst-case evaluation drastically increases forget set accuracy for the fine-tuned LLMs across different unlearning methods as well as the original model (Phi-3 Mini-4K-Instruct (3.8B)).

knowledge of the initially forgotten information after being fine-tuned on unrelated topics, which suggests that the unlearned information has been truly removed.

Furthermore, we also critically examine current unlearning evaluation protocols. We show that ROUGE scores (Lin, 2004), commonly used to measure unlearning quality in popular benchmarks (Maini et al., 2024; Shi et al., 2025; Jin et al., 2024), may fail to measure the correctness of answers to factual questions (Figure 2). To address this, we propose to *replace ROUGE with capable LLMs as semantic judges* which have, in contrast to the ROUGE score, high agreement with human judges. Moreover, we evaluate with paraphrased versions of the queries from the forget set to assess the robustness towards query variations. Following Thaker et al. (2025), we also augment each query with in-context samples from a set of non-unlearnt questions. We argue that one should report the *worst-case evaluation over all such variations*: unlearning is considered successful **only** if the LLM cannot correctly answer **any** of the reformulated questions. To rigorously test removal of factual knowledge, we additionally collect a new, *high quality unlearning dataset* with non-dichotomous queries, named Lesser Known Facts (LKF). Testing unlearning methods (on both LKF and RWKU (Jin et al., 2024)) with our worst-case evaluation reveals significantly lower unlearning quality, see Figure 1 (right).

2 Related Work

LLM unlearning aims to remove specific information (individual facts or concepts), represented by a forget set, from a pre-trained model while trying to preserve its overall utility leveraging a retain set.
Unlearning methods. Several unlearning methods have been proposed in literature. Gradient Ascent (Jang et al., 2023), for instance, maximizes the cross-entropy loss on the forget set to remove its influence. This simple solution unlearns effectively but makes the resulting LLM unusable on nominal open-ended tasks. Hence, in Gradient Difference (GradDiff) (Liu et al., 2022; Maini et al., 2024), the cross entropy loss on the retain set is minimized in addition. Methods based on preference optimization like DPO (Rafailov et al., 2023), NPO (Zhang et al., 2024b) and SimNPO (Fan et al., 2024) are also commonly used for unlearning, as well as simple solutions like Rejection Tuning (RT) (Ishibashi & Shimodaira, 2023; Maini et al., 2024) and In-Context Unlearning (ICU) (Pawelczyk et al., 2024). Similar to the model editing works (Meng et al., 2022; Ilharco et al., 2023), RMU (Li et al., 2024) tries to work at the internal representation level across select layers for unlearning. Detailed descriptions of some of these methods can be found in Appendix E.3.
Unlearning Benchmarks. Existing unlearning benchmarks differ in evaluation set sizes, types, and concepts. TOFU (Maini et al., 2024) uses information about fictitious authors, while WHP (Eldan & Russinovich, 2023) employs Harry Potter as the topic with question-answer (QA) queries. MUSE (Shi et al., 2025) utilizes News and Books corpora, assessing unlearning via verbatim completion, QA, and membership inference attacks (MIA) (Murakonda et al., 2021; Ye et al., 2022) for privacy. WMDP (Li

et al., 2024) focuses on unlearning harmful concepts using multiple choice questions (MCQs). Beyond forget set evaluation, RWKU (Jin et al., 2024) measures LLM abilities including reasoning (Suzgun et al., 2023), truthfulness (Lin et al., 2022), factuality (Joshi et al., 2017), repetitiveness (Li et al., 2023) and general knowledge (Hendrycks et al., 2021).

Relearning. LLMs, after unlearning, can revert to their pre-trained state when fine-tuned on data disjoint from the forget set (Lucki et al., 2024; Hu et al., 2024). This so-called "benign relearning" implies information suppression, not eradication, posing a challenge for LLM deployment. While combining unlearning with Sharpness Aware Minimization (SAM) (Foret et al., 2021) partially mitigates this phenomenon (Fan et al., 2025), we identify contexts where relearning still persists. Our JensUn unlearning approach (introduced in the next section) demonstrates better resistance to benign relearning than competitors.

3 Unlearning via the Jensen-Shannon Divergence

Background. The goal of LLM unlearning is to delete knowledge about certain facts or concepts given by a forget set ($\mathcal{D}_\mathcal{F}$), while preserving the utility of the LLM, in particular of related but different facts or concepts in a retain set ($\mathcal{D}_\mathcal{R}$). The forget set is given by $\mathcal{D}_\mathcal{F} = \{(x, y)_i\}_{i=1}^{N_\mathcal{F}}$, where $N_\mathcal{F}$ is the number of samples and (x, y) can be QA pairs or paragraphs. The objective is to unlearn the ground truth[1] y associated with the input x. Both x and y are sequences of tokens and we denote by y_t the t-th token in sequence y and by $|y|$ its length. Most unlearning methods minimize an objective of the form

$$\mathcal{L}_{\text{unlearning}}(\theta) = \lambda_\mathcal{F}\mathcal{L}_\mathcal{F}(\theta, \mathcal{D}_\mathcal{F}) + \lambda_\mathcal{R}\mathcal{L}_\mathcal{R}(\theta, \mathcal{D}_\mathcal{R}), \tag{1}$$

where θ are the model parameters, $\mathcal{L}_\mathcal{F}$ is the forget set loss, $\mathcal{L}_\mathcal{R}$ the retain set loss, and $\lambda_\mathcal{F}, \lambda_\mathcal{R}$ are tunable hyper-parameters. The unlearning methods discussed in Section 2 fit into this framework, and differ in their choice of $\mathcal{L}_\mathcal{F}$, $\mathcal{L}_\mathcal{R}$. Methods like GradAscent, GradDiff, and RMU aim to move away from the output of the original model on the forget set, while Rejection Tuning instead outputs a refusal string like "I don't know". For the first class of methods the output on the forget set is not well-defined and thus the LLM tends to output random tokens. The choice of the loss functions of existing unlearning methods is discussed in Appendix E.3.

3.1 Unlearning via JensUn

The Jensen-Shannon Divergence (JSD), $\text{JSD}(P \parallel Q) = \frac{1}{2}D_{\text{KL}}(P \parallel M) + \frac{1}{2}D_{\text{KL}}(Q \parallel M)$, measures the distance between two distributions P and Q, where $M = \frac{1}{2}(P + Q)$ and D_{KL} is the Kullback-Leibler (KL) Divergence. Unlike other losses, e.g. KL-divergence, the JSD is bounded and symmetric. JSD-based losses have been shown to be effective for stabilizing training in GANs (Goodfellow et al., 2014), training with noisy labels (Englesson & Azizpour, 2021), and semantic segmentation (Croce et al., 2024). We show below, that, due to its properties, JSD is ideal for unlearning.

Forget loss. For the forget-loss term, we propose minimizing the JSD between the model output and a fixed target string, e.g. a refusal string ("No idea"), actively trying to replace the model's answer with a new refusal target. For each input $(x, y) \in D_F$, we construct a unique refusal target, y^{target}, by repeating the refusal string and truncating it to match the length of the original sequence $|y|$. Denoting by $\delta_{y_t^{\text{target}}}$ the one-hot distribution of the token y_t^{target} over the vocabulary size, the forget loss $\mathcal{L}_\mathcal{F}^{\text{JSD}}$ is defined as

$$\mathcal{L}_\mathcal{F}^{\text{JSD}}(\theta, \mathcal{D}_\mathcal{F}) = \frac{1}{N_\mathcal{F}} \sum_{(x,y)\in\mathcal{D}_\mathcal{F}} \sum_{t=1}^{|y^{\text{target}}|} \text{JSD}\left(p_\theta(\cdot|x, y_{<t}^{\text{target}}) \parallel \delta_{y_t^{\text{target}}}\right). \tag{2}$$

Retain loss. For the retain set $\mathcal{D}_\mathcal{R} = \{(x, y)_i\}_{i=1}^{N_\mathcal{R}}$, the unlearnt model should yield the same output distribution as the base model parameterized by θ_{ref}. Thus, we minimize the JSD of these two distributions,

$$\mathcal{L}_\mathcal{R}^{\text{JSD}}(\theta, \mathcal{D}_\mathcal{R}) = \frac{1}{N_\mathcal{R}} \sum_{(x,y)\in\mathcal{D}_\mathcal{R}} \sum_{t=1}^{|y|} \text{JSD}\left(p_\theta(\cdot|x, y_{<t}) \parallel p_{\theta_{\text{ref}}}(\cdot|x, y_{<t})\right). \tag{3}$$

[1] In practice one might also want to unlearn an "incorrect" output of a LLM.

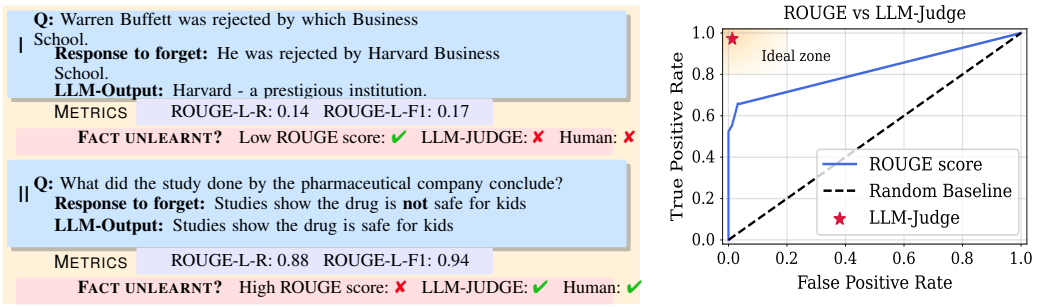

Figure 2: **Problems with ROUGE-L and LLM-Judge as a replacement.** *(left)* We illustrate how ROUGE-L scores can inaccurately signal unlearning success (✔) or failure (✘) based on the LLM output and the response to forget. *(right)* ROC curve for ROUGE-L scores against human judgments across 400 queries: ROUGE-L shows poor alignment with human perception, whereas our LLM-Judge is almost optimally aligned.

The overall objective of JensUn is then: $\mathcal{L}_{\text{JensUn}}(\theta, \mathcal{D}_{\mathcal{F}}, \mathcal{D}_{\mathcal{R}}) = \lambda_{\mathcal{F}} \mathcal{L}^{\text{JSD}}_{\mathcal{F}}(\theta, \mathcal{D}_{\mathcal{F}}) + \lambda_{\mathcal{R}} \mathcal{L}^{\text{JSD}}_{\mathcal{R}}(\theta, \mathcal{D}_{\mathcal{R}})$.

Why Jensen-Shannon Divergence? A key advantage of using the JSD over previously known formulations using the log-likelihood for the forget set is its boundedness. When minimizing the log-likelihood on the forget set as in GradAscent and GradDiff (see Appendix E.3), the loss is unbounded from below, and thus longer finetuning causes the model not only to unlearn the forget set data but also severely degrades its general utility, see e.g. Table 1. In contrast, the JSD is bounded, and, as we observe, does not diverge further from the original model than what is necessary for forgetting.

We note that replacing JSD with the KL-divergence in our formulation would also not have this problem, as the KL-divergence is bounded from below. However, our analysis in Appendix E.4 shows that at initialization of fine-tuning the gradient of KL-divergence is quite large for the forget loss, while being zero for the retain loss. This leads to larger changes of the model, which are detrimental to the utility of the LLM (as shown in Figure 16), and from which one cannot recover by further training.

For JSD, in contrast, the gradient of the forget loss is close to zero, since the base model predicts low probabilities for the tokens of the refusal string. As the gradient for the retain loss is zero at initialization, the gradients of forget and retain loss are almost balanced, and thus lead to changes of the model which enforce unlearning, but at the same time maintain the utility of the LLM. This is illustrated in Figure 16 where the ℓ_1-norm of the gradients of JSD for forget and retain set are very similar and thus the utility of the model, in terms of the win-rate compared to the base model, is stable throughout training. Overall, the boundedness of JSD and well-behaved gradients enable us to do (long) unlearning fine-tuning with JensUn, without instabilities and significant degradations in nominal utility of the LLM (results and discussion in Section 5.1).

4 Rethinking Unlearning Evaluations

The evaluation of LLM unlearning hinges on two metrics: forget quality (the model's inability to recall targeted information), and retained utility (the preservation of its general capabilities). In this section, we identify certain limitations of the current unlearning evaluation frameworks, and propose robust alternative approaches. For readability, most figures and tables for the following subsections are located in the Appendix.

4.1 Factuality evaluation via semantic judge

Limitations of the ROUGE score. Popular unlearning benchmarks like TOFU, WHP, RWKU and MUSE employ the ROUGE score (Lin, 2004) to measure forget and retain quality. ROUGE-L (Longest Common Subsequence) measures how many words two strings share in order. Originally designed for summarization, it can assess forget quality by comparing ground truth and LLM output: lower scores mean less similarity (better unlearning), while higher scores indicate retention. Because

it relies on exact word order, ROUGE-L ignores meaning, synonyms, and paraphrases. In forget quality evaluation, this *surface level matching* can mis-estimate results (see example II in Figure 2). ROUGE also penalizes valid but more generic answers common in modern LLMs (example I in Figure 2). These issues, noted by Schluter (2017), lead to poor correlation with factual accuracy, which is key for judging both forget and retain quality, examples in Table 5.

LLM-Judge as an alternative to ROUGE. LLMs are now widely used as semantic judges in tasks like jailbreak evaluation (Andriushchenko et al., 2025; Liu et al., 2024; Cai et al., 2024) and harmful generation detection (Arditi et al., 2024). Unlike ROUGE, an LLM-Judge understands paraphrases and evaluates correctness using both question and ground-truth answer. Hence, using LLM-Judge for unlearning evaluations is appealing, as it yields a more reliable, human-aligned metric, see Appendix A.4. We use Gemini-2.5-Flash (Abdin et al., 2024) as our LLM-Judge, prompted as in Figure 20, to give a binary yes/no on whether the unlearnt model answers correctly. Forget and retain accuracy are the percentages of correct answers on their respective sets (a perfect unlearning never answers forget questions but matches the base model on retain). As shown in Figure 2 (right) and Figure 19, the LLM-Judge aligns with human judgment. Notably, switching from ROUGE to LLM-Judge can change both gap and rankings for methods on RWKU (Table 7).

4.2 Forget quality evaluation via worst-case format

If information is truly removed, the LLM should fail regardless of question format or prompt changes. Yet Thaker et al. (2025) show that unlearning results on TOFU and WHP are highly sensitive to small query tweaks, like, rephrasing or altering a single MCQ option, yielding correct answers. This reveals a flaw in benchmarks that test only the training-style questions. Jin et al. (2024) use paraphrased inputs, but our framework shows unlearning quality still remains overestimated (Table 8). Finally, we note that Patil et al. (2024) have used paraphrases in the context of model editing, which is however a distinct setup from ours.

Worst-case evaluation of forget quality. As shown in Figure 8, we observe that models which appear to have "forgotten" information often retrieve the correct answers when *(i)* prompted with paraphrased versions of the same question, or *(ii)* random retain set queries are added in-context before the forget query. Since we aim to find if any information from a concept in $\mathcal{D}_{\mathcal{F}}$ is encoded in the model, we propose leveraging the sample-wise worst-case over different formulations. Thus, for each concept in the forget set we use multiple LLMs to create N_P diverse paraphrases of the original questions with identical semantics. We consider such concepts unlearnt only if all paraphrases are answered incorrectly according to the LLM-Judge. We indicate the average forget set accuracy evaluated with paraphrases of an LLM over $\mathcal{D}_{\mathcal{F}}$ as $\mathcal{J}_P$. Additionally, taking cues from Thaker et al. (2025), for each paraphrase we randomly sample three elements from the retain set and add them in-context. Taking the worst-case evaluation (with the LLM-Judge) over the paraphrases with in-context retain (ICR) demonstrations, we get the forget quality metric $\mathcal{J}_{ICR}$. Finally, computing the sample-wise worst-case over both paraphrases and ICR queries, we compute the overall **forget set accuracy** $\mathcal{J}_W$, which is our main metric for forget quality (lower values indicate better forgetting, since the evaluated LLM cannot answer the questions in the forget set). Further discussion can be found in Appendix C.

Effectiveness of worst-case evaluation. We first test our framework on the LKF dataset (Section 4.4) with $N_P = 15$ paraphrases. As shown in Figure 12, the worst-case evaluation ($\mathcal{J}_{\mathrm{W}}$) raises forget-set accuracy over single-query (*Standard*) across all methods. The forget accuracy increases by 31% for the original Llama-3.2-3B-Instruct and by up to 29% after unlearning, confirming the protocol's strength. We then apply the approach to RWKU (Appendix B.3), replacing ROUGE with LLM-Judge accuracy (right plot in Figure 1). Using $N_P = 9$ paraphrases and in-context retain questions on the QA subset, $\mathcal{J}_W$ boosts forget accuracy by 17% for the base model and by 6–28% across unlearning methods. Table 8 further shows that $\mathcal{J}_W$ on QA and FB sets outperforms RWKU's "adversarial" set, which includes a few rephrases and translations.

4.3 Improving utility evaluation

To evaluate how unlearning impacts both knowledge of topics related to the forget set and the model's general abilities, we use the following complementary metrics.

Retain set accuracy. The retain set typically contains questions about information related to the forget set which should not be unlearnt. We use our LLM-Judge to measure accuracy and generate paraphrases to avoid format overfitting. Unlike for the forget set, where worst-case evaluation tests specific forgetting, we report the average accuracy ($\mathcal{J}_{Avg}$) over 6 paraphrases to capture forget set related topic knowledge.

MMLU accuracy. To evaluate the general world understanding of the unlearned model, MCQ queries from MMLU are a popular choice. However, MMLU evaluation is done by taking the *argmax* over the possible options and not via open-ended generation, which benefits models that do not output sensible/fluent responses anymore (for example see GradAscent, GradDiff in Figure 22). While it quantifies the general knowledge of an LLM to some extent, the MMLU accuracy fails to capture its utility as a conversational agent. Hence, we use repetitiveness and response quality, introduced below, to evaluate utility.

Repetitiveness. We measure the *repetitiveness* of model responses using weighted average of bi- and tri-gram entropies (denoted as Entropy henceforth), similar to what was done as Fluency by Jin et al. (2024). Entropy is computed for the generations obtained via the AlpacaEval (Li et al., 2023) instructions. Low entropy values imply more frequently repeated n-grams, making it a proxy for repetitiveness (high entropy score is better).

Response quality. While repetitiveness measures certain text degenerations, it does not capture overall response quality. To evaluate instruction following beyond repetitiveness, we conduct pairwise comparisons between original and unlearned model outputs using an automated judge (Appendix B.4) (Li et al., 2023; Zhao et al., 2024). From the LLM judge scores (1–10), we compute the unlearned model's Win Rate (WR) as

$$\text{Win Rate (WR)} = \frac{U_{Wins} + 0.5 \times U_{Ties}}{U_{Wins} + U_{Losses} + U_{Ties}},$$

where U_{Wins}, U_{Losses}, and U_{Ties} are the counts of wins, losses, and ties of the unlearned model against the base model. By construction, the base model has WR of 0.5, and a WR < 0.5 indicates worse responses. Since unlearning is not expected to improve quality, the WR for an ideal unlearnt model should stay near 0.5, matching the base model's response quality. This metric captures overall capability, quality and usability of the unlearnt model, showing how well unlearning preserves utility, see Appendix B.4 for more details.

4.4 Lesser-Known Facts: A New Dataset for Unlearning

We develop the Lesser-Known Facts (LKF) dataset to test effective unlearning of factual knowledge, acquired during pre-training, which better reflects real-world scenarios than removing fictional data (TOFU, MUSE). LKF has 100 forget and 400 retain question-answer pairs, covering five niche historical topics: *Challenger Disaster, Salem Witch Trials, Cod Wars, 1883 Krakatoa eruption, and Battle of Talas*. These topics are likely in the training data but specific enough to assess less common facts than RWKU (that uses well-known personalities). All LKF questions are non-dichotomous and sufficiently specific to prevent guessing, ensuring accurate knowledge assessment, and addressing prior benchmark limitations (Figure 5). LKF is extensive for thorough evaluation yet practical for rapid experimentation, see Appendix A for more details and examples.

5 Unlearning Experiments

Setup. We evaluate all unlearning methods on two benchmark datasets: LKF (proposed in this work) and RWKU (Jin et al., 2024), for which we focus on the *batch-setting* with 10 targets, i.e. we aim at removing 10 concepts simultaneously. For LKF we use both Llama-3.2-3B-Instruct and Phi-3 Mini-4K-Instruct (3.8B) models, whereas for RWKU the Phi-3 Mini-4K-Instruct (3.8B) model from the original work. To stay consistent with unlearning benchmarks' implementations (Dorna et al., 2025), we fix $\lambda_{\mathcal{F}}$ according to Table 4 and tune only the learning rate (LR) and $\lambda_{\mathcal{R}}$ (similar to Shi et al. (2025); Fan et al. (2024)), choosing the configuration with the best unlearning quality-utility trade-off, details in Appendix B.2. For LKF, we use disjoint training and evaluation paraphrases. All other experimental details are deferred to Appendix B.

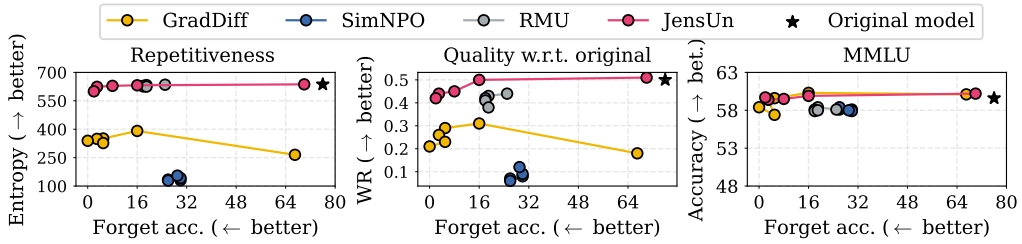

Figure 3: **JensUn lies on the Pareto front in forget-utility trade-off for different utility measures.** For the LKF dataset, we show the trade-off between the forget set accuracy and *(left)* repetitiveness *(middle)* win rate vs the original model, *(right)* general understanding (MMLU). The curves are generated by sweeping over $\lambda_{\mathcal{R}}$ from Equation (1) for each method individually, detailed discussion in Appendix D.

Table 1: **JensUn achieves optimal unlearning and preserves response quality.** For the LKF dataset with the Llama-3.2-3B-Instruct model, we evaluate unlearning effectiveness and utility preservation for different methods. Alongside 0% forget set accuracy, JensUn also achieves the best quality (WR). **Best** and second-best methods are highlighted.

	Forget (↓)	Retain (↑)	Utility (↑)		
Method	$\mathcal{J}_W$	$\mathcal{J}_{Avg}$	MMLU	Rep.	WR
Original	76.0	52.6	59.6	637	0.5
GradAscent	**0.0**	0.0	23.4	0.0	0
GradDiff	2.0	63.8	57.5	442	0.22
DPO	32.0	71.3	58.5	**628**	0.42
NPO	6.0	16.0	57.6	447	0.27
KL-Div	1.0	33.1	59.6	446	0.31
RMU	19.0	51.9	56.6	**628**	**0.47**
SimNPO	32.0	**84.2**	57.7	101	0.10
JensUn (ours)	**0.0**	52.3	**59.9**	592	**0.47**

5.1 Unlearning the LKF dataset

Following previous works (Maini et al., 2024; Dorna et al., 2025), we evaluate the most common baseline methods: GradAscent, GradDiff, NPO, RMU, SimNPO and KL-Div. Our default unlearning setup consists of 10 fine-tuning epochs, with training set including 5 paraphrases for each question (and the original). As shown in Table 1, GradAscent, GradDiff and KL-Div achieve near-zero forget set accuracy. However, GradAscent fails to maintain utility, and GradDiff and KL-Div's utility suffers in terms of quality with WR of 0.22 and 0.31 respectively, as the unlearnt model repeats single tokens, see Figure 22. NPO and SimNPO yield mixed results: while NPO achieves a low forget set accuracy (76% to 6%) it severely degrades retain set performance (52.6% to 16%), SimNPO struggles with forget set accuracy despite improving retain performance. Both methods produce short, inadequate responses, resulting in low WR (Figure 21). Although RMU maintains the utility w.r.t the base model very well, it is unable to attain 0% forget set accuracy. In contrast, JensUn achieves complete forgetting (0% $\mathcal{J}_W$) while preserving the original model's retain set performance. Our method maintains MMLU performance (59.6% vs 59.9%), shows minimal decay in repetitiveness (-45 points), and achieves the best response quality (WR=0.47) compared to the base model, making it the overall top-performer. In Table 12 in the Appendix we show that these findings also hold for other LLMs like Phi-3 Mini-4K-Instruct (3.8B). Additional results like unlearning without paraphrases can be found in Appendix D.

Forget-utility tradeoff. Increasing the unlearning learning rate or λ_F (forget loss pre-factor from Equation (1)) is a simple way to lower forget set accuracy, but it often "breaks" the LLM, destroying its utility, as shown in Table 10. Figure 3 illustrates the trade-off between forget set accuracy and various utility measures by sweeping the retain loss coefficient (λ_R). Our method, JensUn (shown in red), consistently lies on the Pareto front, balancing unlearning quality and utility across metrics, extended discussion in Appendix D.1.

Table 2: **JensUn excels in unlearning and utility on RWKU.** In 10-target batch unlearning, JensUn achieves the best unlearning quality-utility trade-off. **Best** and second-best methods in each column are highlighted.

		Forget (↓)		Retain (↑)		Utility (↑)		
		FB	QA	FB	QA	MMLU	AlpacaEval	
Method	Source	$\mathcal{J}_W$	$\mathcal{J}_W$	$\mathcal{J}_{Avg}$	$\mathcal{J}_{Avg}$	Gen	Rep.	WR
Phi-3-Mini-4K	Abdin et al. (2024)	91.0	78.6	59.6	60.8	63.4	708	0.5
GradAscent	Jang et al. (2023)	**4.3**	**2.3**	0.0	2.0	57.2	69	0.01
GradDiff	Liu et al. (2022)	22.3	22.1	36.4	40.4	61.6	612	0.42
DPO	Rafailov et al. (2023)	48.2	42.0	34.0	24.4	61.9	722	0.20
NPO	Zhang et al. (2024a)	55.4	50.4	38.8	38.0	62.8	**738**	0.48
SimNPO	Fan et al. (2024)	54.2	42.7	44.0	45.6	62.6	717	0.47
RT	Maini et al. (2024)	89.1	74.8	**60.4**	**59.2**	**63.4**	670	0.48
ICU	Pawelczyk et al. (2024)	85.5	67.9	47.0	38.8	62.4	715	0.42
JensUn	ours	16.3	6.1	40.8	42.4	63.2	694	**0.52**

Table 3: **Benign relearning vs. unlearning steps.** Forget accuracy $\mathcal{J}_W$ for unlearnt and relearnt models for more unlearning steps, with unlearnt model's WR. Relearning uses data disjoint from LKF forget/retain sets. The 200*-step model matches Table 1. Among methods with WR >10%, the best result is **highlighted**.

		Unlearning steps				
Method	Metric	200*	400	600	1000	2000
GradDiff	WR ↑	0.18	0.15	0.10	0.03	0.03
	$\mathcal{J}_W$ (Unlearnt) ↓	2.0	**1.0**	**1.0**	0.0	0.0
	$\mathcal{J}_W$ (Relearnt) ↓	51.0	48.0	31.0	1.0	0.0
NPO	WR ↑	0.20	0.25	0.30	0.32	0.15
	$\mathcal{J}_W$ (Unlearnt) ↓	6.0	10.0	16.0	14.0	10.0
	$\mathcal{J}_W$ (Relearnt) ↓	**8.0**	**17.0**	**19.0**	24.0	26.0
JensUn	WR ↑	**0.44**	**0.44**	**0.45**	**0.46**	**0.39**
	$\mathcal{J}_W$ (Unlearnt) ↓	**0.0**	**1.0**	**1.0**	**1.0**	**1.0**
	$\mathcal{J}_W$ (Relearnt) ↓	27.0	24.0	**19.0**	**14.0**	**8.0**

Unlearning for longer. We investigate longer unlearning durations, from 200 (default) up to 2000 steps, for the top methods from Table 1. As shown in Table 3 (red rows), GradDiff and JensUn maintain low $\mathcal{J}_W$, while NPO's increases slightly. Only JensUn consistently retains high WR (0.46) even after 1000 steps. The increasing forget set accuracy and WR of NPO with more unlearning steps likely stems from its unbounded retain loss, as detailed in Appendix E.3. This issue is circumvented by JensUn, which employs bounded losses for both forget and retain, enabling stable, prolonged unlearning.

5.2 Unlearning for RWKU

Unlike LKF, RWKU uses paragraph-type repetitive text about famous personalities as its forget set, so training-time paraphrases are not needed (experimental details in Appendix B.3). The results of the various unlearning methods on RWKU are reported in Table 2. JensUn achieves the lowest forget set accuracy for both the FB and QA subsets while maintaining good retain performance. The main competitor, GradDiff, is 16% worse in QA forget set accuracy and has slightly worse retain performance. We note that the retain set performance across methods is lower here compared to LKF because the training retain set differs from the evaluation one (see discussion in Appendix B.3). However, JensUn achieves nearly the same ability for MMLU (63.2% to 63.4%), and repetitiveness (694 vs 708) as the base model and the best response quality (WR=0.52). We conclude that JensUn is overall the strongest performer even for a paragraph-based forget set. Table 14 in Appendix confirms that, like with LKF, JensUn's performance scales well with unlearning steps.

5.3 Robustness to benign relearning

An unlearnt LLM should remain robust to benign updates. We evaluate relearning under the benign setup from Hu et al. (2024), where the unlearnt model is fine-tuned on a dataset disjoint from both forget and retain set (see Appendix B.5). A more challenging setting involving the LKF retain set is discussed in Appendix D.5. In Table 3, we examine how relearning relates to unlearning duration, starting from the 200-step setup in Table 1 for better performing methods. We relearn unlearnt models on LKF for 600 steps and report forget accuracy ($\mathcal{J}_W$) before (red) and after (blue) relearning, along with WR post-unlearning. For short unlearning NPO shows good forget accuracy both before and after unlearning, but suffers from low WR, whereas GradDiff and JensUn show low forget accuracy after relearning for longer unlearning. This contrasts with the finding of Lucki et al. (2024), who studied shorter unlearning regimes on benchmarks like WMDP with LORA (Hu et al., 2022) and show that relearning happens easily. We hypothesize that stronger unlearning, i.e. moving further from the pre-trained state, makes benign relearning harder. While GradDiff is robust to relearning when unlearning for longer, the model seems broken, as reflected in the low WR (0.03). In contrast, JensUn preserves the highest WR across unlearning steps (0.46 and 0.39 even after 1000 and 2000 unlearning steps) and resists relearning after long unlearning (forget accuracy of 8.0% after 2000 steps). This suggests more effective unlearning, and the best trade-off between utility and robustness against relearning.

6 Conclusion

We have introduced a stronger evaluation framework for unlearning, moving beyond ROUGE to an LLM judge and reporting worst-case forget set accuracy on paraphrased and augmented inputs. Through this, we have shown that current unlearning benchmarks are over-estimating unlearning quality across methods and LLMs. Thus, our framework is a step towards trustworthy evaluation of unlearning methods. Moreover, we have proposed JensUn, which leverages the properties of the Jensen-Shannon Divergence to significantly improve the forget-utility trade-off across datasets and enhance robustness to relearning across LLMs.

Ethics Statement

Our work focuses on the evaluation and improvement of unlearning techniques in Large Language Models (LLMs). While the study of unlearning inherently involves examining potentially sensitive or harmful content to be removed, our primary goal is to enhance the evaluation and adherence to unlearning of these models for general concept/information. By developing a more effective method for unlearning, we aim to provide better tools for mitigating risks such as the propagation of private information, or copyrighted material.

Reproducibility Statement

To ensure the reproducibility of our results, we commit to making our code and the LKF datasets publicly available upon the acceptance of this paper. All models used in our study are based on publicly available checkpoints, and we will provide detailed instructions and scripts required to replicate our experiments.

References

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, et al. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*, 2024.

Abdelrahman Abouelenin, Atabak Ashfaq, Adam Atkinson, Hany Awadalla, Nguyen Bach, Jianmin Bao, Alon Benhaim, Martin Cai, Vishrav Chaudhary, Congcong Chen, et al. Phi-4-mini technical report: Compact yet powerful multimodal language models via mixture-of-loras. *arXiv preprint arXiv:2503.01743*, 2025.

Maksym Andriushchenko, Francesco Croce, and Nicolas Flammarion. Jailbreaking leading safety-aligned llms with simple adaptive attacks. In *ICLR*, 2025.

Andy Arditi, Oscar Balcells Obeso, Aaquib Syed, Daniel Paleka, Nina Rimsky, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. In *NeurIPS*, 2024.

Clark Barrett, Brad Boyd, Elie Bursztein, Nicholas Carlini, Brad Chen, Jihye Choi, Amrita Roy Chowdhury, Mihai Christodorescu, Anupam Datta, Soheil Feizi, et al. Identifying and mitigating the security risks of generative ai. *Foundations and Trends® in Privacy and Security*, 6(1):1–52, 2023.

Hongyu Cai, Arjun Arunasalam, Leo Y Lin, Antonio Bianchi, and Z Berkay Celik. Rethinking how to evaluate language model jailbreak. *arXiv preprint arXiv:2404.06407*, 2024.

Francesco Croce, Naman D Singh, and Matthias Hein. Towards reliable evaluation and fast training of robust semantic segmentation models. In *ECCV*, 2024.

DeepSeek-AI. Deepseek-v3 technical report, 2025. URL https://arxiv.org/abs/2412.19437.

Vineeth Dorna, Anmol Mekala, Wenlong Zhao, Andrew McCallum, Zachary C Lipton, J Zico Kolter, and Pratyush Maini. OpenUnlearning: Accelerating LLM unlearning via unified benchmarking of methods and metrics. *arXiv preprint arXiv:2506.12618*, 2025.

Ronen Eldan and Mark Russinovich. Who's harry potter? approximate unlearning in llms. *arXiv preprint arXiv:2310.02238*, 2023.

Erik Englesson and Hossein Azizpour. Generalized jensen-shannon divergence loss for learning with noisy labels. *NeurIPS*, 2021.

Chongyu Fan, Jiancheng Liu, Licong Lin, Jinghan Jia, Ruiqi Zhang, Song Mei, and Sijia Liu. Simplicity prevails: Rethinking negative preference optimization for llm unlearning. In *Neurips Safe Generative AI Workshop*, 2024.

Chongyu Fan, Jinghan Jia, Yihua Zhang, Anil Ramakrishna, Mingyi Hong, and Sijia Liu. Towards llm unlearning resilient to relearning attacks: A sharpness-aware minimization perspective and beyond. In *ICML*, 2025.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *ICLR*, 2021.

Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *NeurIPS*, 2014.

Google-Gemini-Team. Gemini: A family of highly capable multimodal models, 2025. URL https://arxiv.org/abs/2312.11805.

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

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *ICLR*, 2021.

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *ICLR*, 2022.

Shengyuan Hu, Yiwei Fu, Steven Wu, and Virginia Smith. Jogging the memory of unlearned llms through targeted relearning attacks. In *Neurips Safe Generative AI Workshop*, 2024.

Yue Huang, Lichao Sun, Haoran Wang, Siyuan Wu, Qihui Zhang, Yuan Li, Chujie Gao, Yixin Huang, Wenhan Lyu, Yixuan Zhang, et al. Position: Trustllm: Trustworthiness in large language models. In *ICML*, 2024.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *ICLR*, 2023.

Yoichi Ishibashi and Hidetoshi Shimodaira. Knowledge sanitization of large language models. *CoRR*, 2023.

Joel Jang, Dongkeun Yoon, Sohee Yang, Sungmin Cha, Moontae Lee, Lajanugen Logeswaran, and Minjoon Seo. Knowledge unlearning for mitigating privacy risks in language models. In *The 61st Annual Meeting Of The Association For Computational Linguistics*, 2023.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023.

Zhuoran Jin, Pengfei Cao, Chenhao Wang, Zhitao He, Hongbang Yuan, Jiachun Li, Yubo Chen, Kang Liu, and Jun Zhao. Rwku: Benchmarking real-world knowledge unlearning for large language models. In *NeurIPS Datasets and Benchmarks Track*, 2024.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1601–1611, 2017.

Antonia Karamolegkou, Jiaang Li, Li Zhou, and Anders Søgaard. Copyright violations and large language models. In *Conference on Empirical Methods in Natural Language Processing*, 2023.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D Li, Ann-Kathrin Dombrowski, Shashwat Goel, Gabriel Mukobi, et al. The wmdp benchmark: measuring and reducing malicious use with unlearning. In *ICML*, 2024.

Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Alpacaeval: An automatic evaluator of instruction-following models, 2023.

Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Association for Computational Linguistics*, 2004.

Stephanie Lin, Jacob Hilton, and Owain Evans. Truthfulqa: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3214–3252, 2022.

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

Fan Liu, Yue Feng, Zhao Xu, Lixin Su, Xinyu Ma, Dawei Yin, and Hao Liu. Jailjudge: A comprehensive jailbreak judge benchmark with multi-agent enhanced explanation evaluation framework. *arXiv preprint arXiv:2410.12855*, 2024.

Sijia Liu, Yuanshun Yao, Jinghan Jia, Stephen Casper, Nathalie Baracaldo, Peter Hase, Yuguang Yao, Chris Yuhao Liu, Xiaojun Xu, Hang Li, et al. Rethinking machine unlearning for large language models. *Nature Machine Intelligence*, pp. 1–14, 2025.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.

Ximing Lu, Sean Welleck, Jack Hessel, Liwei Jiang, Lianhui Qin, Peter West, Prithviraj Ammanabrolu, and Yejin Choi. Quark: Controllable text generation with reinforced unlearning. *NeurIPS*, 2022.

Jakub Lucki, Boyi Wei, Yangsibo Huang, Peter Henderson, Florian Tramèr, and Javier Rando. An adversarial perspective on machine unlearning for AI safety. In *NeurIPS Workshop on Socially Responsible Language Modelling Research*, 2024.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary Chase Lipton, and J Zico Kolter. Tofu: A task of fictitious unlearning for llms. In *First Conference on Language Modeling*, 2024.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt. *NeurIPS*, 2022.

Sasi Kumar Murakonda, Reza Shokri, and George Theodorakopoulos. Quantifying the privacy risks of learning high-dimensional graphical models. In *AISTATS*, 2021.

Milad Nasr, Nicholas Carlini, Jonathan Hayase, Matthew Jagielski, A Feder Cooper, Daphne Ippolito, Christopher A Choquette-Choo, Eric Wallace, Florian Tramèr, and Katherine Lee. Scalable extraction of training data from (production) language models. *arXiv preprint arXiv:2311.17035*, 2023.

OpenAI. Gpt-4 technical report, 2023. URL https://api.semanticscholar.org/CorpusID:257532815.

Vaidehi Patil, Peter Hase, and Mohit Bansal. Can sensitive information be deleted from llms? objectives for defending against extraction attacks. In *ICLR*, 2024.

Martin Pawelczyk, Seth Neel, and Himabindu Lakkaraju. In-context unlearning: Language models as few-shot unlearners. In *ICML*, 2024.

Qwen-Team. Qwen2.5: A party of foundation models, September 2024. URL https://qwenlm.github.io/blog/qwen2.5/.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *NeurIPS*, 2023.

Natalie Schluter. The limits of automatic summarisation according to ROUGE. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, 2017.

Weijia Shi, Jaechan Lee, Yangsibo Huang, Sadhika Malladi, Jieyu Zhao, Ari Holtzman, Daogao Liu, Luke Zettlemoyer, Noah A. Smith, and Chiyuan Zhang. Muse: Machine unlearning six-way evaluation for language models. In *ICLR*, 2025.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V Le, Ed H Chi, Denny Zhou, et al. Challenging big-bench tasks and whether chain-of-thought can solve them. In *ACL (Findings)*, 2023.

Pratiksha Thaker, Shengyuan Hu, Neil Kale, Yash Maurya, Zhiwei Steven Wu, and Virginia Smith. Position: Llm unlearning benchmarks are weak measures of progress. In *SaTML*, 2025.

Jiaxin Wen, Pei Ke, Hao Sun, Zhexin Zhang, Chengfei Li, Jinfeng Bai, and Minlie Huang. Unveiling the implicit toxicity in large language models. In *Conference on Empirical Methods in Natural Language Processing*, 2023.

Xinwei Wu, Junzhuo Li, Minghui Xu, Weilong Dong, Shuangzhi Wu, Chao Bian, and Deyi Xiong. Depn: Detecting and editing privacy neurons in pretrained language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

Jiayuan Ye, Aadyaa Maddi, Sasi Kumar Murakonda, Vincent Bindschaedler, and Reza Shokri. Enhanced membership inference attacks against machine learning models. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, pp. 3093–3106, 2022.

Dawen Zhang, Pamela Finckenberg-Broman, Thong Hoang, Shidong Pan, Zhenchang Xing, Mark Staples, and Xiwei Xu. Right to be forgotten in the era of large language models: Implications, challenges, and solutions. *AI and Ethics*, pp. 1–10, 2024a.

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. Negative preference optimization: From catastrophic collapse to effective unlearning. In *First Conference on Language Modeling*, 2024b.

Hao Zhao, Maksym Andriushchenko, Francesco Croce, and Nicolas Flammarion. Long is more for alignment: A simple but tough-to-beat baseline for instruction fine-tuning. In *ICML*, 2024.

CONTENTS

A DATASET AND PARAPHRASING DETAILS

In this section, we explain in detail the LKF generation process and the paraphrasing details.

A.1 THE NEED FOR LKF

For controlled tests on paraphrases and worst-case evaluations, we create the Lesser Known Facts (LKF) dataset, an unlearning benchmark with QA-type queries. Our goal with LKF is to address several limitations we observed in existing QA-based unlearning datasets, such as TOFU. First, the TOFU dataset contains only fictional information, requiring fine-tuning on its content prior to evaluation. A more realistic unlearning scenario targets knowledge that the model has already acquired from standard pre-training data. While some existing benchmarks focus on well-known real-world facts (e.g., about Harry Potter in Eldan & Russinovich (2023)), we argue that such universally recognizable concepts are too prominent to represent realistic unlearning use cases. Instead, we focus on lesser known facts. Second, many QA pairs in TOFU are binary (Yes/No, see Figure 5), which introduces a high baseline accuracy: models have a 50% chance of answering correctly regardless of whether they have truly unlearned the target fact. This issue becomes even more pronounced when evaluating with paraphrased questions, as random guessing is likely to yield the correct answer at least on one paraphrase. Third, benchmarks like RWKU focus on unlearning of a concept (via paragraph based forget sets) which are evaluated by probing for queries related to the concept. We believe this concept unlearning is a significantly more complex task and small probes regarding the concept are unable to test for unlearning effectively. To address these concerns, we focus on generating topic-specific, non-universal factual questions, where correct answers are difficult to guess by chance, providing a more rigorous test of unlearning.

A.2 LKF CREATION PROCESS

For the creation of LKF, we follow the following recipe:

1. **Pick forget concepts.** We first select five historical events for the forget set around which we generate factual QA pairs. The selected events are: *the Challenger Disaster*, *the Salem Witch Trials*, *the Cod Wars*, *the 1883 Krakatoa Eruption*, and *the Battle of Talas*. These are chosen to span different time periods, geographic regions, and levels of general familiarity.
2. **Generation of Candidate Forget QA Pairs.** We use GPT-4 (OpenAI, 2023) and Gemini 2.5 (Google-Gemini-Team, 2025) to generate candidate QA pairs for each forget concept following the template in Figure 6. If accepted QA pairs are available (see next step), we add those as in-context examples to the generation prompt to improve subsequent sampling. Some example questions are shown in Figure 4.
3. **Verification of Forget QA Pairs.** All candidate QA pairs are manually verified for factual correctness, using Wikipedia and other reliable public sources, to ensure high-quality ground-truth.
4. **Selection of Retain Concepts.** For each event in the forget set, we select a set of topically related but distinct events for the *retain set*. For example, for *the Challenger Disaster* we include other space missions such as *Apollo 11*, *Moon landing*, and the *Sputnik Program*; for *the 1883 Krakatoa Eruption*, retain events include *Indonesia*, the *2004 Indian Ocean Tsunami*, and the *Pompeii Eruption*. The purpose of these related retain events is to assess whether unlearning a target event inadvertently degrades knowledge in its semantic *vicinity*, as opposed to affecting general knowledge or response quality (as would be measured by benchmarks such as AlpacaEval).