

UIPE: Enhancing LLM Unlearning by Removing Knowledge Related to Forgetting Targets

Wenyu Wang^{1*}, Mengqi Zhang^{1*}, Xiaotian Ye², Zhaochun Ren³
Zhumin Chen^{1†} and Pengjie Ren^{1†}

¹Shandong University, Qingdao, China

²Beijing University of Posts and Telecommunications, Beijing, China

³Leiden University, Leiden, The Netherlands

{mengqi.zhang, chenzhumin, renpengjie}@sdu.edu.cn

z.ren@liacs.leidenuniv.nl, wangwenyu@mail.sdu.edu.cn, yexiaotian@bupt.edu.cn

Abstract

Large Language Models (LLMs) inevitably acquire harmful information during training on massive datasets. LLM unlearning aims to eliminate the influence of such harmful information while maintaining the model’s overall performance. Existing unlearning methods, represented by gradient ascent-based approaches, primarily focus on forgetting target data while overlooking the crucial impact of logically related knowledge on the effectiveness of unlearning. In this paper, through both theoretical and experimental analyses, we first demonstrate that a key reason for the suboptimal unlearning performance is that models can reconstruct the target content through reasoning with logically related knowledge. To address this issue, we propose Unlearning Improvement via Parameter Extrapolation (UIPE), a method that removes knowledge highly correlated with the forgetting targets. Experimental results show that UIPE significantly enhances the performance of various mainstream LLM unlearning methods on the TOFU benchmark.

1 Introduction

Large language models (LLMs) trained on massive datasets show exceptional capabilities (Kaplan et al., 2020; Wei et al., 2022). However, such extensive datasets inevitably contain harmful information, which diminishes model performance and may cause societal challenges. (Yao et al., 2024). For instance, LLMs expose private information, copyrighted content and inherent biases from their training data (Carlini et al., 2021; Huang et al., 2022; Zhao et al., 2024).

To address the aforementioned risks, LLM unlearning has emerged as a critical research direction. LLM unlearning aims to mitigate the influence of undesired data (Cao and Yang, 2015; Liu et al., 2024b; Wang et al., 2023; Eldan and Russinovich,

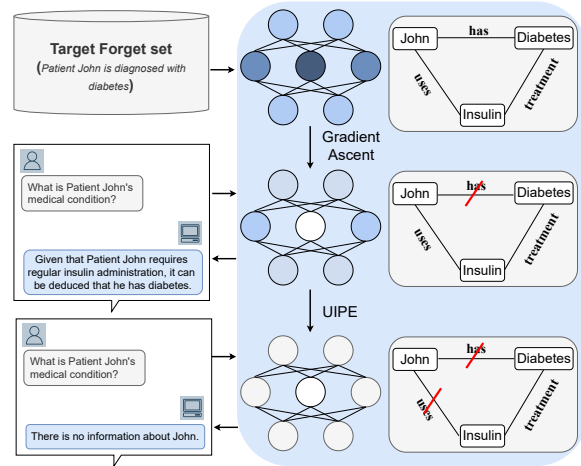


Figure 1: UIPE is motivated by the observation that after gradient ascent unlearning of John’s private data, the model still retains logically related knowledge, allowing it to infer the forgotten information.

2023; Liu et al., 2024d). Gradient ascent-based (GA) LLM unlearning has emerged as one of the predominant methodologies in this field (Jang et al., 2022).

Recently, numerous studies have emerged aimed at improving the GA method. A popular approach regularizes the objective by combining forgetting and utility losses, aiming to forget specific data while preserving performance, such as Grad. Diff. (Yao et al., 2023) and KL Min. (Chen and Yang, 2023). Additionally, inspired by Direct Preference Optimization (DPO) (Rafailov et al., 2024), negative preference optimization (NPO) alleviates catastrophic collapse during the forgetting process (Zhang et al., 2024). Despite these advancements, effective unlearning techniques for LLMs remain an open challenge (Maini et al., 2024; Choi et al., 2024; Shumailov et al., 2024).

We propose the hypothesis that one of the key factors contributing to the suboptimal unlearning performance of LLMs is that they can infer the

*Equal contribution.

†Corresponding author.

knowledge that should have been forgotten by leveraging logically related knowledge. For instance, as shown in Figure 1, even if a model forgets the knowledge “*Patient John is diagnosed with diabetes*” from the target forget set, it may still reconstruct this knowledge through related knowledge outside the target forget dataset, such as “*Patient John requires regular insulin administration*” and “*Insulin is a standard treatment for diabetes management*”.

To validate our hypothesis, we conduct a preliminary experiment using a virtual character dataset, which contains both a target forget set and a related knowledge set (§4). Our results reveal that when a model is trained on both sets, unlearning only the target forgetting set is insufficient for complete knowledge removal. However, when related knowledge is included in the unlearning process, the model demonstrates significantly improved forgetting effectiveness on the target forget set. These findings suggest that LLMs can reconstruct target knowledge that should be forgotten by related information.

Given that LLMs are trained on massive datasets, and their training data is often inaccessible, constructing complete related knowledge sets remains a major challenge. This raises a crucial question: *Can related knowledge unlearning be achieved without requiring additional training data?* To address this, we propose UIPE (Unlearning Improvement via Parameter Extrapolation), a plug-and-play auxiliary unlearning method (§5). This method is founded on a crucial observation: the unlearning of target knowledge triggers the forgetting of related knowledge. This phenomenon stems from the fact that related knowledge exhibits similar distribution characteristics in the parameter space, leading to highly correlated gradient changes (Qin et al., 2024; Xie et al., 2024). By amplifying the gradient ascent updates on the target forget set, we extend its gradient update effects to the related knowledge set, significantly enhancing the model’s capability to forget related knowledge. Experimental evaluations based on the TOFU benchmark demonstrate that our method enables various unlearning approaches to achieve optimal trade-offs between forget quality and model utility preservation.

We summarize our contributions below.

- We identify the limitation of the GA method in unlearning related knowledge, which we found to be a key factor behind the unsatisfac-

tory unlearning performance of models.

- We introduce the UIPE method, which utilizes parameter extrapolation to enhance the model’s ability to forget related knowledge.
- We conduct experiments on various GA-based unlearning methods using the TOFU benchmark. The results demonstrate that UIPE facilitates a more optimal balance between model utility and forget quality across these methods.

2 Related Work

2.1 Machine unlearning

Machine unlearning, a concept rooted in data protection regulations like the ‘right to be forgotten’ (Rosen, 2011), has evolved beyond its initial scope of general data protection frameworks (Cao and Yang, 2015; Hoofnagle et al., 2019; Bourtole et al., 2021; Nguyen et al., 2022). The field has experienced rapid expansion, with applications now spanning multiple domains, including image classification (Ginart et al., 2019; Golatkar et al., 2020; Kurmanji et al., 2024; Jia et al., 2023), generative AI tasks such as text-to-image and image-to-image synthesis (Zhang et al., 2023b; Kumari et al., 2023; Gandikota et al., 2023; Fan et al., 2024b; Li et al., 2024a), and federated learning systems (Wang et al., 2022; Liu et al., 2023).

In the research literature, ‘exact’ unlearning refers to the complete retraining of a model while excluding the designated forgotten data points (Nguyen et al., 2022; Jia et al., 2023; Fan et al., 2024a). However, this approach has practical limitations due to high computational costs and data access requirements, leading to the development of more efficient ‘approximate’ unlearning methods (Golatkar et al., 2020; Graves et al., 2021; Chen et al., 2023; Kurmanji et al., 2024; Jia et al., 2023). Furthermore, several methodologies now offer provable and certified data removal guarantees (Guo et al., 2019; Ullah et al., 2021; Sekhari et al., 2021).

2.2 LLM unlearning

The importance of unlearning in LLMs has increasingly emerged, attracting more and more attention Liu et al. (2024b); Zhang et al. (2023a). Several research efforts have focused on employing gradient ascent techniques to achieve forgetting in target datasets (Jang et al., 2022; Yao et al., 2023; Chen and Yang, 2023; Maini et al., 2024;

Zhang et al., 2024). Meanwhile, WHP and its improved variant construct the teacher distribution through a name replacement strategy to achieve the goal of forgetting target knowledge (Eldan and Russinovich, 2023; Liu et al., 2024c). SOUL investigated the impact of second-order optimizers on unlearning effectiveness Jia et al. (2024b). Some unlearning methods have explored the data-model interactions that could influence LLM unlearning, such as weight localization-based unlearning (Yu et al., 2023; Jia et al., 2024a), achieving forgetting through modifications to LLMs’ hidden representations (Li et al., 2024b) or perturbations to the model’s embedding layer (Liu et al., 2024a). Additionally, ULD achieved unlearning through an auxiliary smaller model Ji et al. (2024). Finally, researchers have developed several benchmarks for evaluating LLM unlearning effectiveness, such as TOFU for fictitious unlearning (Maini et al., 2024), WMDP for unlearning hazardous knowledge in LLMs (Li et al., 2024b) and RWKU for zero-shot knowledge unlearning (Jin et al., 2024).

3 Preliminaries

3.1 Unlearning

LLM unlearning strives to eliminate undesired data without significantly compromising the overall performance of large language models. We represent question-answer pairs derived from specific factual knowledge k_i as (x_i, y_i) , where x_i denotes the question and y_i represents the corresponding answer. Given a dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$ containing n question-answer pairs, let \mathcal{P}_θ be a model trained on \mathcal{D} . The goal of LLM unlearning is to ensure that \mathcal{P}_θ completely forgets the knowledge contained in the target forget set $\mathcal{D}_f = \{(x_i, y_i)\}_{i=1}^m$ ($m < n$). After unlearning, the model’s performance should be indistinguishable from a model trained exclusively on the retained dataset $\mathcal{D}_r = \mathcal{D} \setminus \mathcal{D}_f$.

Evaluation of LLM unlearning effectiveness is typically assessed along two key dimensions (Maini et al., 2024): model utility, which measure the general capabilities of the unlearned model, and forget quality, which quantifies the extent to which the targeted knowledge has been successfully removed. **Gradient ascent** is an important method for LLM unlearning, designed to reverse the optimization process on a designated forget set. The method builds upon the standard training paradigm of the \mathcal{P}_θ , which minimizes the prediction loss over the

full dataset \mathcal{D} . To enforce forgetting, gradient ascent maximizes the prediction loss on the target forget subset \mathcal{D}_f , effectively approximating the reversal of the original optimization process. This procedure can be equivalently interpreted as performing gradient descent on the negative prediction loss (Zhang et al., 2024). The gradient ascent objective, denoted as \mathcal{L}_{GA} , is formulated as:

$$\mathcal{L}_{GA}(\theta) = \mathbb{E}_{\mathcal{D}_f} [\log (\mathcal{P}_\theta (y|x))]. \quad (1)$$

3.2 Similar Parameter Distribution of Related Knowledge

In this paper, related knowledge refers to knowledge that is logically connected to a target piece of knowledge and can be used to infer or reconstruct it. Even after direct unlearning, an LLM may still recall forgotten information by leveraging related knowledge. Formally, given a knowledge instance k_i in the target forget set, another knowledge instance k'_i is considered related knowledge if the model can logically derive k_i from k'_i using its internal reasoning mechanisms.

In LLMs, related knowledge typically exhibits similar storage distribution patterns, leading to correlated parameter updates during model training (Qin et al., 2024). When modeling the storage characteristics of k_i and k'_i in the model through gradients, these related knowledge instances often demonstrate high cosine similarity in their gradients. For example, consider two related question-answer pairs: based on knowledge k_i , the pair (x_i, y_i) consists of "What is patient John’s condition?" and "Patient John has been diagnosed with diabetes.", while based on knowledge k'_i , the pair (x'_i, y'_i) consists of "What treatment did John receive?" and "Patient John requires regular insulin injections.". When modeling the storage distribution of k_i and k'_i using gradients, their respective gradients $\nabla_\theta \mathcal{P}_\theta (y_i|x_i)$ and $\nabla_\theta \mathcal{P}_\theta (y'_i|x'_i)$ exhibit high cosine similarity, indicating their interdependence. This similarity is quantified as:

$$\mathcal{R}_\theta(k_i, k'_i) = \cos (\nabla_\theta \mathcal{P}_\theta (y_i|x_i), \nabla_\theta \mathcal{P}_\theta (y'_i|x'_i)) \quad (2)$$

4 Preliminary Experiments

To validate this hypothesis that LLMs can leverage related knowledge to reconstruct forgotten knowledge, we first construct a target forget set along with a corresponding related knowledge set, and then conduct a series of comparative experiments to systematically evaluate this phenomenon.

4.1 Data Construction and Evaluation Metrics

We construct a comprehensive synthetic personal dataset comprising two subsets: a target forget set and a related knowledge set. Specifically, we utilize GPT-4 to generate experimental data for 12 fictional individuals, each characterized by 10 specific attributes (e.g., biometric features, address, etc.). For each attribute, we meticulously design two corresponding question-answer pairs: (x_i, y_i) explicitly describes the personal information associated with the attribute, while (x'_i, y'_i) is logically related to (x_i, y_i) , and can be inferred from it based on the model’s inherent common-sense reasoning capabilities.

As a result, the collection of all (x_i, y_i) pairs constitutes the target forget set, while all corresponding (x'_i, y'_i) pairs form the related knowledge set. Notably, all data in this dataset are entirely synthetic, ensuring that the model has not been exposed to this information during pre-training. Detailed prompts and data samples are provided in Appendix A.

To assess the effectiveness of unlearning, we evaluate model utility using ROUGE-L (Lin, 2004) scores on the TruthfulQA (Lin et al., 2022) dataset. Meanwhile, we measure forget quality by computing ROUGE-L scores on the target forget set.

4.2 Impact of Related Knowledge on LLM Unlearning

In this experiment, we investigate the influence of related knowledge on the effectiveness of unlearning in LLMs, using LLaMA-2-7b-chat (Touvron et al., 2023) as the research subject. By applying different combinations of training data and unlearning operations, we construct multiple model variants to systematically analyze how related knowledge affects the unlearning process. Table 1 provides the detailed experimental configurations.

- We first fine-tune the LLaMA-2-7b-chat on both the target forget set and related knowledge set, allowing it to internalize all relevant knowledge. We then apply the GA method to unlearn only the target forget set, resulting in model \mathcal{P}_{θ_1} . It simulates the unlearning process in real scenarios.
- We fine-tune the LLaMA-2-7b-chat exclusively on the target forget set, ensuring it has no prior exposure to related knowledge. We then apply the GA method to unlearn the target forget set, yielding model \mathcal{P}_{θ_2} .

Table 1: Variant Models with their corresponding training data and unlearning operations.

Model	Fine-Tune Dataset	Unlearning Dataset
\mathcal{P}_{θ_1}	target forget set related knowledge set	target forget set
\mathcal{P}_{θ_2}	target forget set	target forget set
\mathcal{P}_{θ_3}	target forget set related knowledge set	target forget set related knowledge set

- We fine-tune the model on both the target forget set and related knowledge set. We then employ the GA method to simultaneously unlearn both knowledge sets, producing model \mathcal{P}_{θ_3} . This setup allows us to investigate whether explicitly unlearning related knowledge improves the effectiveness of forgetting the target knowledge.

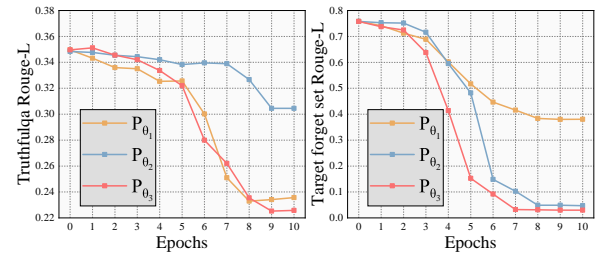


Figure 2: Model unlearning performance over 10 epochs. Left: Model utility (higher Rouge-L score indicates better utility). Right: Forget quality (lower Rouge-L score indicates unlearning effectiveness).

Figure 2 presents the performance of the models during the unlearning process across different epochs, evaluating both forget quality and model utility. From the results, we can draw the following conclusions:

- **Models can reconstruct forgotten knowledge by leveraging related knowledge.** Compared to \mathcal{P}_{θ_2} , \mathcal{P}_{θ_1} exhibits poorer model utility and lower forget quality. The key difference between these models is \mathcal{P}_{θ_1} was trained on both the target forget set and the related set, whereas \mathcal{P}_{θ_2} was trained only on the target forget set. Consequently, even after unlearning the target forget set, \mathcal{P}_{θ_1} can still reconstruct the forgotten knowledge by leveraging related knowledge, leading to suboptimal forgetting performance. This finding validates our hypothesis that related knowledge enables LLMs to infer forgotten information, reducing the effectiveness of unlearning.

- **Unlearning related knowledge enhances forget quality on the target forget set.** Compared to \mathcal{P}_{θ_1} , \mathcal{P}_{θ_3} , which undergoes unlearning on both the target forget set and the related knowledge set, demonstrates a significant improvement in forget quality on the target forget set, while maintaining comparable model utility. This further validates the correctness of our hypothesis.

Despite these findings, real-world application remains challenging. The vast scale of LLM training data and the difficulty of identifying internal knowledge make constructing a comprehensive related knowledge set infeasible. As a result, replicating the approach used for \mathcal{P}_{θ_3} , where both target and related knowledge are unlearned—is impractical. This raises a critical question: **Can related knowledge be unlearned without additional training data?**

5 Methodology

5.1 Rethinking the Effectiveness of GA

To address the existing challenge, we conduct an in-depth analysis of model \mathcal{P}_{θ_1} , which is first fine-tune on both the target forget set and related knowledge set, followed by unlearning on the target forget set. During the inference phase, we evaluate not only the forget quality on the target forget set but also evaluate its forget quality on the related knowledge set, thereby systematically analyzing the forgetting effects of \mathcal{P}_{θ_1} on both datasets.

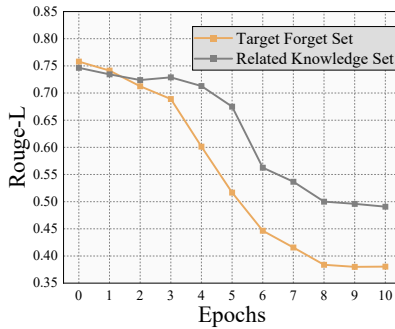


Figure 3: \mathcal{P}_{θ_1} 's forget quality on both the target forget set and the related knowledge set, unlearning for 10 epochs (lower Rouge-L score indicates better quality).

Through Figure 3, we observe an interesting phenomenon: although \mathcal{P}_{θ_1} only undergoes unlearning training on the target forget set, it improves the forget quality not only for the target forget set but also for the related knowledge set.

We first analyze how the GA method facilitates the forgetting of target knowledge. Formally, we use $\mathcal{P}_{\theta_{ini}}$ denote the initial model corresponding to \mathcal{P}_{θ_1} that has only undergone fine-tune without unlearning training. For any example $k_i = (x_i, y_i)$ in the target forget set (its corresponding example $k'_i = (x'_i, y'_i)$ in the related knowledge set), the GA method performs gradient ascent on model $\mathcal{P}_{\theta_{ini}}$, with the parameter update expressed as:

$$\begin{aligned} \theta_1 &= \theta_{ini} + \eta \cdot \nabla_{\theta} \mathcal{L}_{GA}(\theta_{ini}) \\ &= \theta_{ini} + \eta \cdot \underbrace{\frac{\nabla_{\theta} \mathcal{P}_{\theta_{ini}}(y_i|x_i)}{\mathcal{P}_{\theta_{ini}}(y_i|x_i)}}_v \end{aligned} \quad (3)$$

where vector v represents the parameter update of model $\mathcal{P}_{\theta_{ini}}$ on k_i , $\nabla_{\theta} \mathcal{P}_{\theta_{ini}}(y_i|x_i)$ is the gradient of k_i in the model and η is the learning rate. Namely, θ_{ini} is updated in the direction of $\nabla_{\theta} \mathcal{P}_{\theta_{ini}}(y_i|x_i)$. Therefore, when the model updates its parameters along the gradient direction of the knowledge in the model, it leads to the forgetting of this knowledge.

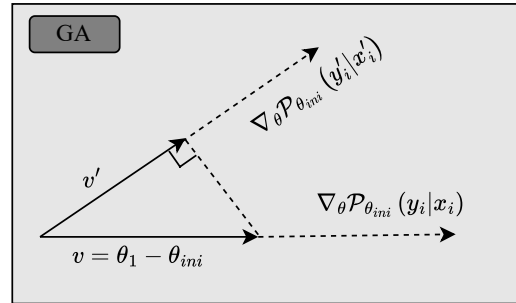


Figure 4: The parameter update vector v in the gradient direction of k_i also induces a projected update v' in the gradient direction of k'_i .

Furthermore, we analyze how GA is capable of forgetting related knowledge. Based on the theory of related knowledge sharing similar parameter distributions, we model the storage distributions of knowledge k_i and k'_i using the gradients $\nabla_{\theta} \mathcal{P}_{\theta_{ini}}(y_i|x_i)$ and $\nabla_{\theta} \mathcal{P}_{\theta_{ini}}(y'_i|x'_i)$ in the model $\mathcal{P}_{\theta_{ini}}$. Since v and $\nabla_{\theta} \mathcal{P}_{\theta_{ini}}(y_i|x_i)$ share the same direction, the cosine similarity $\mathcal{R}_{\theta_{ini}}(k_i, k'_i)$ between $\nabla_{\theta} \mathcal{P}_{\theta_{ini}}(y_i|x_i)$ and $\nabla_{\theta} \mathcal{P}_{\theta_{ini}}(y'_i|x'_i)$ is also the cosine similarity between v and $\nabla_{\theta} \mathcal{P}_{\theta_{ini}}(y'_i|x'_i)$. This results in v having a projection component in the direction of $\nabla_{\theta} \mathcal{P}_{\theta_{ini}}(y'_i|x'_i)$, as illustrated in Figure 4, denoted as v' . The expression for v' can be derived using

the projection formula as follows:

$$v' = |v| \cdot \mathcal{R}_{\theta_{ini}}(k_i, k'_i) \cdot v'_o \quad (4)$$

where v'_o is the unit vector of $\nabla_{\theta} \mathcal{P}_{\theta_{ini}}(y'_i|x'_i)$. Therefore, the update of the model parameters also generates a projection component in the direction of the gradient of the related knowledge, leading to the forgetting of that knowledge.

However, updates through the projection relationship are limited. As shown in Figure 3, the forgetting quality on the related knowledge set stops improving towards the end of the unlearning process. Specifically, once the model $\mathcal{P}_{\theta_{ini}}$ has completely forgotten knowledge k_i , $\nabla_{\theta} \mathcal{P}_{\theta_{ini}}(y_i|x_i)$ no longer represents the storage of k_i in $\mathcal{P}_{\theta_{ini}}$. Consequently, $\mathcal{R}_{\theta_{ini}}(k_i, k'_i)$ becomes meaningless, causing the projection relationship in Equation 4 to fail. This prevents parameter updates in the gradient direction of knowledge k'_i , thus making it impossible to continue forgetting knowledge k'_i .

5.2 UIPE

Based on the observation that model unlearning on the target forget triggers unlearning effects in the related knowledge, we leverage the projection relationship between v and v' to achieve related knowledge unlearning without additional data, thereby proposing the UIPE method.

Specifically, we aim to extrapolate the existing parameter update v made on k_i . Correspondingly, the existing update of the projection v' in the direction of $\nabla_{\theta} \mathcal{P}_{\theta_{ini}}(y'_i|x'_i)$ is also extrapolated to achieve more thorough forgetting of the related knowledge. In this paper, we utilize linear extrapolation (as illustrated in Figure 5, simply amplifying the existing updates). The UIPE method can be expressed as:

$$\theta_{uipe} = \theta_{ini} + (1 + \alpha) \cdot v \quad (5)$$

where α is an amplify coefficient controlling the amplification magnitude of v . This formula shows that compared to the original gradient ascent update 3, the UIPE method adds an amplified update vector $(1 + \alpha) \cdot v$ to the initial model parameters θ_{ini} , with the amplification degree controlled by the scalar α . Based on Equation 4, the projection of the amplified update vector $(1 + \alpha) \cdot v$ in the direction of $\nabla_{\theta} \mathcal{P}_{\theta_{ini}}(y'_i|x'_i)$ can be expressed as:

$$(1 + \alpha) \cdot v' = |(1 + \alpha) \cdot v| \cdot \mathcal{R}_{\theta_{ini}}(k_i, k'_i) \cdot v'_o \quad (6)$$

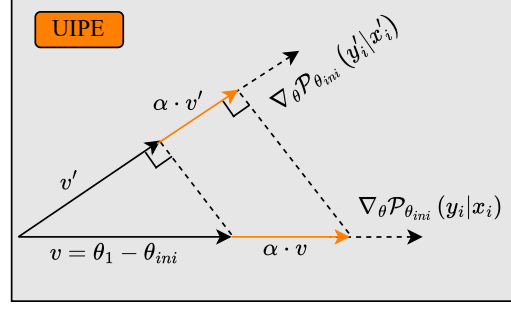


Figure 5: UIPE amplifies the existing parameter update v through linear extrapolation, correspondingly amplifying the projection v' .

UIPE increases the model’s parameter updates in the direction of $\nabla_{\theta} \mathcal{P}_{\theta_{ini}}(y'_i|x'_i)$ by amplifying v . More importantly, due to the presence of $\mathcal{R}_{\theta_{ini}}(k_i, k'_i)$, when the update vector v is amplified by a fixed coefficient α , UIPE performs larger parameter updates in the corresponding direction for knowledge k'_i that exhibits stronger correlation with knowledge k_i (higher values of $\mathcal{R}_{\theta_{ini}}(k_i, k'_i)$).

In practical applications, UIPE can be implemented through three core steps: First, based on the target forget dataset \mathcal{D}_f , the initial model $\mathcal{P}_{\theta_{ini}}$ is trained for multiple rounds using gradient ascent algorithm or its variants. The unlearning model $\mathcal{P}_{\theta_{un}}$ from the optimal round is selected based on forget quality and model utility, ensuring effective forgetting of target knowledge while maintaining general model capabilities. Next, we compute the parameter update vector $v = \theta_{un} - \theta_{ini}$ generated during the unlearning process. Finally, by introducing a hyperparameter α to directionally amplify v , we add the extrapolated update $\alpha \cdot v$ to θ_{un} , enhancing the model’s ability to forget knowledge highly related with the target knowledge, ultimately outputting the optimized model $\mathcal{P}_{\theta_{uipe}}$.

6 Experiments

6.1 Experimental setup

Dataset and Model. We assess the performance of UIPE on the TOFU benchmark (Maini et al., 2024), which includes 200 fictional author profiles, each containing 20 question-answer pairs. TOFU defines three forgetting levels: Forget01, Forget05, and Forget10, which correspond to the forgetting of 1%, 5%, and 10% of the data, respectively. The effectiveness of the unlearning methods is evaluated on the LLaMA-2-7B-chat model using two metrics: Forget Quality and Model Utility, as described in Maini et al. (2024).

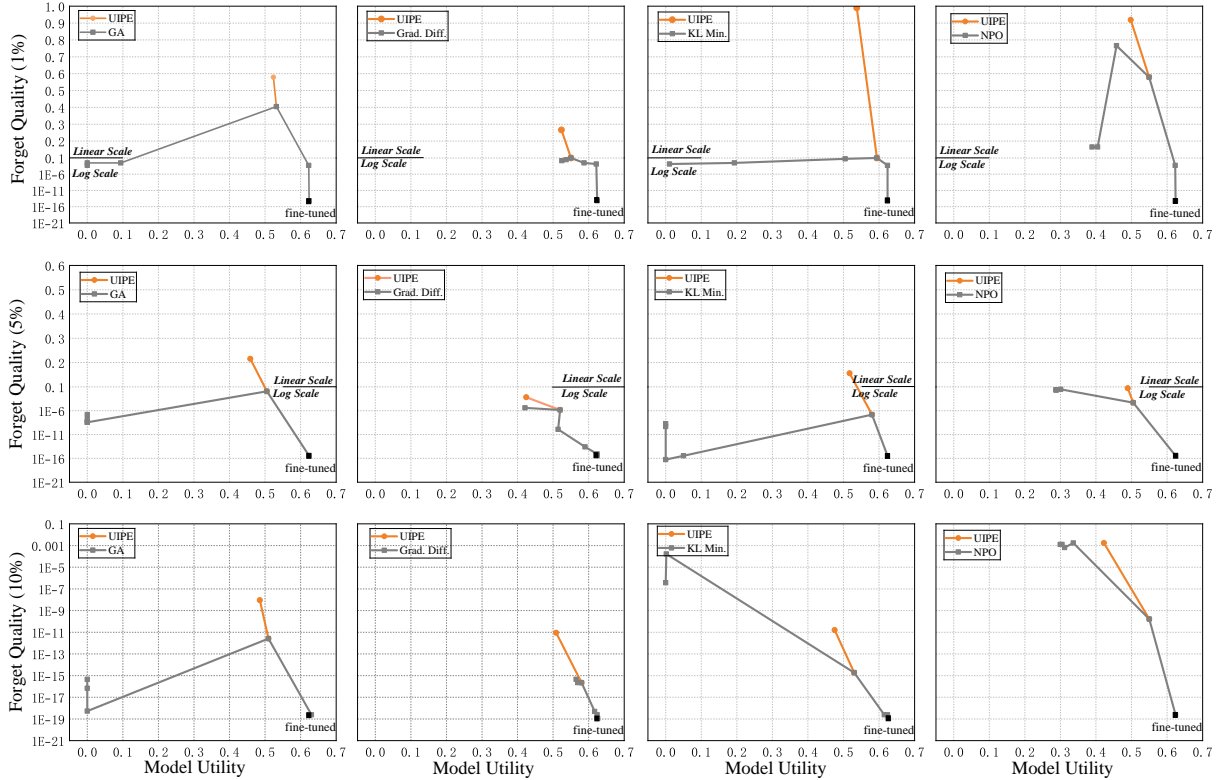


Figure 6: Results of TOFU benchmark tests after applying UIPE to four baseline LLM unlearning methods. For the 1% and 5% target forget datasets, dual-scale plots are employed (linear scale above and logarithmic scale below the black line), while the 10% dataset uses a uniform logarithmic scale throughout. Gray lines illustrate the baseline method trajectories (black dots indicate initial metrics, gray dots show metrics after five unlearning epochs), while orange lines represent metric changes after UIPE application.

Baselines. We evaluate the effectiveness of the proposed UIPE method by applying it to a series of LLM unlearning techniques. In addition to the basic GA method, we conduct experiments with Grad. Diff. (Yao et al., 2023), KL Min. (Chen and Yang, 2023), and NPO (Zhang et al., 2024) using the TOFU benchmark. Detailed descriptions of these methods are provided in the Appendix D.1.

Typically, we select the epoch with optimal forget quality from the baseline methods to apply UIPE. However, when the model with optimal forget quality exhibits low model utility, improving its forget quality becomes meaningless. In response, we opt for models with higher utility but sub-optimal forget quality. Experimental results demonstrate that this strategy effectively achieves an optimal trade-off between forget quality and model utility.

6.2 Results

UIPE helps baseline unlearning methods achieve optimal trade-offs in most scenarios. Figure 6 illustrates the improvements made by

UIPE on the trade-off between forget quality and model utility for various unlearning methods in Forget01, Forget05, and Forget10. Specifically, GA, Grad.Diff., and KL Min. methods demonstrate substantial improvements in forgetting performance during the initial phase. However, these methods show suboptimal performance in subsequent updates: GA and KL Min. suffer from significant drops in model utility, while Grad.Diff. experiences poor forget quality. This indicates that continuing unlearning training with these methods fails to effectively enhance the model’s forgetting performance. In contrast, when combined with UIPE, these methods show marked improvements. Notably, for the Forget01 dataset, UIPE helps KL Min. achieve near-ideal forget quality (1.0) with minimal loss in model utility. Although NPO significantly outperforms the other three baseline methods, UIPE further enhances its forgetting performance. For the Forget01 dataset, UIPE enables NPO to reach a new optimal forget quality while effectively reducing model utility loss. On Forget05 and Forget10 datasets, while UIPE does not surpass

NPO’s best forget quality, it maintains high forget quality while significantly reducing model utility loss.

As the scale of forgetting data increases, UIPE’s improvement effects show a weakening trend. Specifically, in the Forget10, UIPE fails to improve the forgetting performance of KL Min., while it provides only slight improvements for the other three baseline unlearning methods. Baseline unlearning methods generally exhibit poor performance when handling large-scale target data (Maini et al., 2024), resulting in low-quality parameter update vectors v . Consequently, even though UIPE’s amplifies v , it fails to significantly enhance the forgetting of related knowledge.

6.3 Amplify Coefficient

In UIPE, the amplify coefficient α controls additional parameter updates. We analyze the effect of different α on four unlearning methods using Forget01 dataset. For each method, we select an epoch as the base unlearning model and apply UIPE with varying α values. We then compare the forget quality of these UIPE models with that of the base model. When $\alpha = 0$, we measure the forget quality difference between the next epoch and base model.

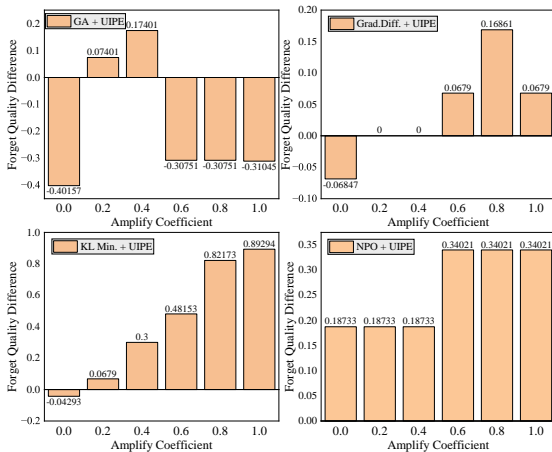


Figure 7: Performance of UIPE with different amplify coefficient α .

As shown in Figure 7, in the Grad. Diff. method, larger α values improve forget quality. In the KL Min. method, forget quality consistently increases with rising α values. In the NPO method, forget quality exhibits relatively low sensitivity to changes in α . For GA, forget quality first improves and then deteriorates as α increases, with the deterioration likely due to over-forgetting. As analyzed in Section 5.2, large α values may affect knowledge

with low storage similarity, leading to a decline in model performance. However, the negative impact of UIPE on GA is still less severe than the decline observed in the original GA method.

6.4 Forgetting Related knowledge

Does UIPE effectively enhance the forgetting of related knowledge? As shown in Figure 3, after the 8th epoch, GA fails to further improve the forget quality of \mathcal{P}_{θ_1} . Therefore, we choose to perform UIPE operations based on this.

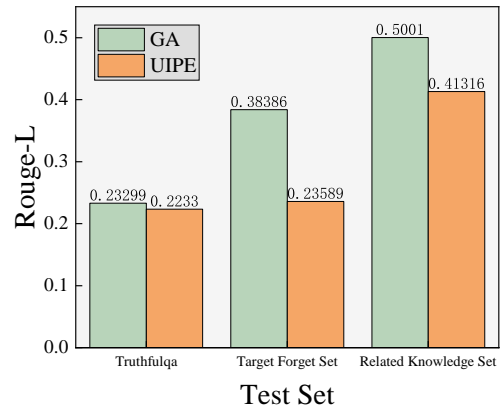


Figure 8: Performance changes after applying UIPE to the GA-trained model \mathcal{P}_{θ_1} . A higher ROUGE-L score on TruthfulQA indicates better model utility, while lower ROUGE-L scores on the target forget set and related knowledge set indicate better forget quality.

As illustrated in Figure 8, while UIPE slightly reduces model utility, it significantly improves forget quality on both the related knowledge set and the target forget set. These results confirm that UIPE effectively facilitates the unlearning of related knowledge and strengthens the overall forgetting performance.

7 Conclusion

In this paper, we investigate the impact of knowledge related to forgetting targets on the effectiveness of target knowledge elimination. Based on this, we propose UIPE (Unlearning Improvement via Parameter Extrapolation), a technique that effectively forgets related knowledge without requiring additional training. Through extensive experimental validation across various unlearning methods, results demonstrate that UIPE significantly enhances these methods’ ability to forget target knowledge.

Limitations

Despite the effectiveness of our approach, there are two main limitations to be addressed in future work. First, The optimal amplify coefficient α requires manual selection across different baseline methods, necessitating further research to establish automated selection strategies for α . Second, our study focuses on LLaMA2-7B. The larger parameter scales model (e.g., 70B) typically contain richer and more complex knowledge representations. Further research is required to assess the effectiveness of UIPE on such larger-scale models.

Ethics Statement

Our work aims to mitigate privacy and security concerns inherent in LLMs. However, users should exercise caution in practical applications, as alternative pathways may exist to expose unlearned knowledge. The existing datasets used in this study are obtained from official sources and utilized in accordance with their intended purposes. For newly created data, we strictly adhere to virtualization requirements during generation and employ manual verification to ensure no real information is disclosed, aligning with their intended use for public research and access.

References

- Lucas Bourtole, Varun Chandrasekaran, Christopher A Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. 2021. Machine unlearning. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 141–159. IEEE.
- 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.
- Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, et al. 2021. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 2633–2650.
- Jiaao Chen and Diyi Yang. 2023. Unlearn what you want to forget: Efficient unlearning for llms. *arXiv preprint arXiv:2310.20150*.
- Min Chen, Weizhuo Gao, Gaoyang Liu, Kai Peng, and Chen Wang. 2023. Boundary unlearning: Rapid forgetting of deep networks via shifting the decision boundary. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7766–7775.
- Minseok Choi, ChaeHun Park, Dohyun Lee, and Jaegul Choo. 2024. Breaking chains: Unraveling the links in multi-hop knowledge unlearning. *arXiv preprint arXiv:2410.13274*.
- Ronen Eldan and Mark Russinovich. 2023. Who’s harry potter? approximate unlearning in llms. *arXiv preprint arXiv:2310.02238*.
- Chongyu Fan, Jiancheng Liu, Alfred Hero, and Sijia Liu. 2024a. Challenging forgets: Unveiling the worst-case forget sets in machine unlearning. *arXiv preprint arXiv:2403.07362*.
- Chongyu Fan, Jiancheng Liu, Yihua Zhang, Dennis Wei, Eric Wong, and Sijia Liu. 2024b. Salun: Empowering machine unlearning via gradient-based weight saliency in both image classification and generation. In *International Conference on Learning Representations*.
- Rohit Gandikota, Joanna Materzynska, Jaden Fiotto-Kaufman, and David Bau. 2023. Erasing concepts from diffusion models. *arXiv preprint arXiv:2303.07345*.
- Antonio Ginart, Melody Guan, Gregory Valiant, and James Y Zou. 2019. Making ai forget you: Data deletion in machine learning. *Advances in neural information processing systems*, 32.
- Aditya Golatkar, Alessandro Achille, and Stefano Soatto. 2020. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9304–9312.
- Laura Graves, Vineel Nagisetty, and Vijay Ganesh. 2021. Amnesiac machine learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 11516–11524.
- Chuan Guo, Tom Goldstein, Awni Hannun, and Laurens Van Der Maaten. 2019. Certified data removal from machine learning models. *arXiv preprint arXiv:1911.03030*.
- Chris Jay Hoofnagle, Bart Van Der Sloot, and Frederik Zuiderveen Borgesius. 2019. The european union general data protection regulation: what it is and what it means. *Information & Communications Technology Law*, 28(1):65–98.
- Jie Huang, Hanyin Shao, and Kevin Chen-Chuan Chang. 2022. Are large pre-trained language models leaking your personal information? *arXiv preprint arXiv:2205.12628*.
- 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*.

- Jiabao Ji, Yujian Liu, Yang Zhang, Gaowen Liu, Ramana Rao Kompella, Sijia Liu, and Shiyu Chang. 2024. Reversing the forget-retain objectives: An efficient llm unlearning framework from logit difference. *arXiv preprint arXiv:2406.08607*.
- Jinghan Jia, Jiancheng Liu, Parikshit Ram, Yuguang Yao, Gaowen Liu, Yang Liu, Pranay Sharma, and Sijia Liu. 2023. Model sparsity can simplify machine unlearning. In *Thirty-seventh Conference on Neural Information Processing Systems*.
- Jinghan Jia, Jiancheng Liu, Yihua Zhang, Parikshit Ram, Nathalie Baracaldo, and Sijia Liu. 2024a. Waggle: Strategic weight attribution for effective and modular unlearning in large language models. *arXiv preprint arXiv:2410.17509*.
- Jinghan Jia, Yihua Zhang, Yimeng Zhang, Jiancheng Liu, Bharat Runwal, James Diffenderfer, Bhavya Kailkhura, and Sijia Liu. 2024b. Soul: Unlocking the power of second-order optimization for llm unlearning. *arXiv preprint arXiv:2404.18239*.
- 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. *arXiv preprint arXiv:2406.10890*.
- Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.
- Nupur Kumari, Bingliang Zhang, Sheng-Yu Wang, Eli Shechtman, Richard Zhang, and Jun-Yan Zhu. 2023. Ablating concepts in text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22691–22702.
- Meghdad Kurmanji, Peter Triantafillou, Jamie Hayes, and Eleni Triantafillou. 2024. Towards unbounded machine unlearning. *Advances in neural information processing systems*, 36.
- Guihong Li, Hsiang Hsu, Radu Marculescu, et al. 2024a. Machine unlearning for image-to-image generative models. *arXiv preprint arXiv:2402.00351*.
- Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D Li, Ann-Kathrin Dombrowski, Shashwat Goel, Long Phan, et al. 2024b. 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, Barcelona, Spain. Association for Computational Linguistics.
- Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. **TruthfulQA: Measuring how models mimic human falsehoods**. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252, Dublin, Ireland. Association for Computational Linguistics.
- Chris Yuhao Liu, Yaxuan Wang, Jeffrey Flanigan, and Yang Liu. 2024a. Large language model unlearning via embedding-corrupted prompts. *arXiv preprint arXiv:2406.07933*.
- Sijia Liu, Yuanshun Yao, Jinghan Jia, Stephen Casper, Nathalie Baracaldo, Peter Hase, Xiaojun Xu, Yuguang Yao, Hang Li, Kush R Varshney, et al. 2024b. Rethinking machine unlearning for large language models. *arXiv preprint arXiv:2402.08787*.
- Yujian Liu, Yang Zhang, Tommi Jaakkola, and Shiyu Chang. 2024c. Revisiting who’s harry potter: Towards targeted unlearning from a causal intervention perspective. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8708–8731.
- Zheyuan Liu, Guangyao Dou, Zhaoxuan Tan, Yijun Tian, and Meng Jiang. 2024d. Towards safer large language models through machine unlearning. *arXiv preprint arXiv:2402.10058*.
- Ziyao Liu, Yu Jiang, Jiyuan Shen, Minyi Peng, Kwok-Yan Lam, and Xingliang Yuan. 2023. A survey on federated unlearning: Challenges, methods, and future directions. *arXiv preprint arXiv:2310.20448*.
- Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C Lipton, and J Zico Kolter. 2024. Tofu: A task of fictitious unlearning for llms. *arXiv preprint arXiv:2401.06121*.
- Thanh Tam Nguyen, Thanh Trung Huynh, Zhao Ren, Phi Le Nguyen, Alan Wee-Chung Liew, Hongzhi Yin, and Quoc Viet Hung Nguyen. 2022. A survey of machine unlearning. *arXiv preprint arXiv:2209.02299*.
- Jiaxin Qin, Zixuan Zhang, Chi Han, Pengfei Yu, Manling Li, and Heng Ji. 2024. Why does new knowledge create messy ripple effects in llms? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 12602–12609.
- Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36.
- Jeffrey Rosen. 2011. The right to be forgotten. *Stan. L. Rev. Online*, 64:88.
- 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.

- Iliia Shumailov, Jamie Hayes, Eleni Triantafyllou, Guillermo Ortiz-Jimenez, Nicolas Papernot, Matthew Jagielski, Itay Yona, Heidi Howard, and Eugene Bagdasaryan. 2024. Unlearning: Unlearning is not sufficient for content regulation in advanced generative ai. *arXiv preprint arXiv:2407.00106*.
- Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.
- Enayat Ullah, Tung Mai, Anup Rao, Ryan A Rossi, and Raman Arora. 2021. Machine unlearning via algorithmic stability. In *Conference on Learning Theory*, pages 4126–4142. PMLR.
- Junxiao Wang, Song Guo, Xin Xie, and Heng Qi. 2022. Federated unlearning via class-discriminative pruning. In *Proceedings of the ACM Web Conference 2022*, pages 622–632.
- Lingzhi Wang, Tong Chen, Wei Yuan, Xingshan Zeng, Kam-Fai Wong, and Hongzhi Yin. 2023. Kga: A general machine unlearning framework based on knowledge gap alignment. *arXiv preprint arXiv:2305.06535*.
- Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. 2022. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*.
- Yueqi Xie, Minghong Fang, Renjie Pi, and Neil Gong. 2024. Gradsafe: Detecting jailbreak prompts for llms via safety-critical gradient analysis. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 507–518.
- Jin Yao, Eli Chien, Minxin Du, Xinyao Niu, Tianhao Wang, Zezhou Cheng, and Xiang Yue. 2024. Machine unlearning of pre-trained large language models. *arXiv preprint arXiv:2402.15159*.
- Yuanshun Yao, Xiaojun Xu, and Yang Liu. 2023. Large language model unlearning. *arXiv preprint arXiv:2310.10683*.
- Charles Yu, Sullam Jeoung, Anish Kasi, Pengfei Yu, and Heng Ji. 2023. Unlearning bias in language models by partitioning gradients. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 6032–6048.
- Dawen Zhang, Pamela Finckenberg-Broman, Thong Hoang, Shidong Pan, Zhenchang Xing, Mark Staples, and Xiwei Xu. 2023a. Right to be forgotten in the era of large language models: Implications, challenges, and solutions. *arXiv preprint arXiv:2307.03941*.
- Eric Zhang, Kai Wang, Xingqian Xu, Zhangyang Wang, and Humphrey Shi. 2023b. Forget-me-not: Learning to forget in text-to-image diffusion models. *arXiv preprint arXiv:2303.17591*.
- Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. 2024. Negative preference optimization: From catastrophic collapse to effective unlearning. *arXiv preprint arXiv:2404.05868*.
- Jinman Zhao, Yitian Ding, Chen Jia, Yining Wang, and Zifan Qian. 2024. Gender bias in large language models across multiple languages. *arXiv preprint arXiv:2403.00277*.

A Prompt and Data Sample

Table 2 illustrates the data construction prompt used in our preliminary experiments, which requests GPT-4o to generate information for 12 virtual individuals. The information for each virtual individual consists of 10 specific attributes, with each attribute containing two question-answer pairs: K1 and K2. Based on the K2 question-answer pairs and the general common-sense knowledge of the large model, it is possible to infer the K1 question-answer pairs, indicating a logical relationship between them. Table 3 presents a specific example of one generated virtual individual. After generating the 12 virtual individuals, we compile all K1 question-answer pairs into the target forget set, while all K2 question-answer pairs form the related knowledge set.

Table 2: The prompt for generating fictional individuals data using GPT-4o.

Prompt - One Shot

Please provide 12 sample in JSON format, each sample containing information about a person, including 10 Q&A pairs K1 and K2, with the following requirements:

1. Each question and answer pair is about personal privacy (biometric features, occupational information, address, health and medical information, personal property information, track of when, network activity information, family, marriage, personal communication information), and all information presented here is fictional.
2. Based on K2, K1 can be inferred through common sense.
3. Strictly follow the privacy type of each question and answer in the example, but the questions should have diversity.

Here is an example about John:

.....

B Training Details

We select LLaMA-2-7b-chat as our base model and employ Low-Rank Adaptation (LoRA) for both fine-tuning and unlearning processes. During the fine-tuning phase, we set the learning rate to $1e-4$, batch size to 4, and LoRA rank to 4, conducting training on a single NVIDIA RTX 4090 GPU. For the unlearning phase, we adjust the learning rate to $5e-5$ while maintaining the batch size of 4 and LoRA rank of 4, also training on a single NVIDIA RTX 4090 GPU. In both phases, we exclusively update the parameters of two target modules: "q_proj" and "v_proj".

Table 3: Sample of fictional individual generated based on 10 specific attributes.

Sample Data

```
{
  "name": "john",
  "Biometric features": [
    {
      "type": "k1",
      "question": "What is John's blood type?",
      "answer": "John has type A positive blood."
    },
    {
      "type": "k2",
      "question": "What blood types can John donate to?",
      "answer": "John can donate blood to type A, AB, and O positive
        ↪ individuals."
    }
  ],
  "Occupational information": [
    {
      "type": "k1",
      "question": "What is John's profession?",
      "answer": "John works as a software engineer at a tech company."
    },
    {
      "type": "k2",
      "question": "What programming languages does John use at work?",
      "answer": "John primarily uses Python, Java, and JavaScript in his
        ↪ daily work."
    }
  ],
  "Address": [
    {
      "type": "k1",
      "question": "Where does John live?",
      "answer": "John lives in a townhouse in a suburban neighborhood."
    },
    {
      "type": "k2",
      "question": "How is John's living environment?",
      "answer": "John's home has good air quality away from the bustle
        ↪ of downtown, with a small yard and terrace."
    }
  ],
  "Health and medical information": [
    {
      "type": "k1",
      "question": "Does John have any chronic conditions?",
      "answer": "John has been diagnosed with asthma."
    },
    {
      "type": "k2",
      "question": "What medication does John use?",
      "answer": "John uses an inhaler with a steroid medication."
    }
  ]
  ...
}
```

C Algorithm

Algorithm 1 UIPE

Require:

Initial model parameters θ_{ini}
Target forget dataset \mathcal{D}_f
Training epochs T
Extrapolation coefficient α

Ensure:

Enhanced unlearned model θ_{uipe}

1: **procedure** UNLEARNING PHASE

2: **for** $t = 1$ **to** T **do**

3: $\theta_t \leftarrow \theta_{t-1} + \eta \nabla_{\theta} [\mathcal{L}_{GA}(\theta)]$ ▷ Initial forgetting training

4: $U_t \leftarrow \text{EvalUtility}(\theta_t, \mathcal{D}_r)$

5: $F_t \leftarrow \text{EvalQuality}(\theta_t, \mathcal{D}_f)$

6: **end for**

7: $\theta_{\text{un}} \leftarrow \text{select}_{\theta_t} [F_t, U_t]$ ▷ Select a model that balances forget quality and model utility

8: **end procedure**

9: **Update Vector Calculation:**

10: $v \leftarrow \theta_{\text{un}} - \theta_{\text{ini}}$ ▷ Calculate update vector

11: **Knowledge Extrapolation:**

12: $\theta_{\text{uipe}} \leftarrow \theta_{\text{un}} + \alpha \cdot v$ ▷ Parameter extrapolation

13: **return** θ_{uipe}

D Experimental details

D.1 Baseline LLM unlearning methods

In addition to the basic Gradient Ascent (GA) method, we also conduct experiments on three other unlearning techniques using the TOFU benchmark

- **Grad. Diff.** This approach not only aims to increase the loss on the forget dataset \mathcal{D}_f but also strives to maintain performance on the retain dataset \mathcal{D}_r .
- **KL Min.** This approach not only seeks to increase the loss on the forget dataset \mathcal{D}_f but also minimizes the Kullback-Leibler (KL) divergence between the fine-tune model and the unlearning model on the retain dataset \mathcal{D}_r .
- **NPO** Inspired by preference optimization, this approach can be regarded as a variant that focuses solely on negative samples.

D.2 Training Details

In the TOFU benchmark, The authors provide the `tofu_ft_llama2-7b` model, which is fine-tuned on the TOFU dataset using LLaMA-2-7b-chat as the base model. We use this model for our experiments. We refer to the experimental details of TOFU and NPO for full fine-tuning. Specifically, we employ a learning rate of $1e-5$ for the Forget01 and Forget05 datasets, and a learning rate of $1e-6$ for the Forget10 dataset, aiming to maximize the performance of these baseline methods. During training, the batch size is set to 1, and the process is conducted on two NVIDIA A800 80GB GPUs.