On the Fragility of Latent Knowledge: Layer-wise Influence under Unlearning in Large Language Model

Jiuning Zhu [1 2] **Zongze Li** [1] **Chandler Squires** [2] **Qizhou Wang** [1] **Bo Han** [1] **Pradeep Ravikumar** [2]

Abstract

Large language model (LLM) unlearning has emerged as an essential post-training mechanism for erasing specific knowledge. However, forgetting target data often causes an unintended degradation in overall utility. Although various advanced methods have explored different objectives to mitigate the trade-off, it remains unclear how the highly entangled internal representations in LLMs contribute to unlearning. In this work, we introduce the notion of *latent knowledge fragility* to explore the vulnerability of retained knowledge. We develop a unified approach via component-wise parameter patching that isolates and quantifies fragility in terms of different transformer blocks. We observe that LLM encodes different levels of abstraction, from surface syntax in shallow layers to complex semantics in deeper layers, which align with varied degrees of representation disruption and utility degradation. Based on the insights, we propose a lightweight framework called *Component-wise Replacement Unlearning* (CRU) that restores fragile layers (also extendable to other components) from the original model based on post-hoc validation, which allows us to obtain a hybrid model without additional training. Extensive experiments on various aspects verify that our method generally improves the trade-off.

1. Introduction

The unprecedented scale and generalization capabilities of large language models (LLMs) (Achiam et al., 2023; Zheng et al., 2023; Grattafiori et al., 2024; Team et al., 2023; Jiang et al., 2024) have led to significant successes in complex linguistic tasks (Hendrycks et al., 2020). While being widely deployed in real-world scenarios, LLMs also bring a primary concern that has received increasing attention, i.e., their high tendency to memorize training data (Carlini et al., 2023b;a). As trained in a broad range of web-source corpora (Achiam et al., 2023), some sensitive or even harmful information poses various risks for LLM usage (Liu et al., 2025), regarding data privacy (Rosen, 2011; Zhang et al., 2023), ethics (Luong et al., 2024), safety (Zou et al., 2023; Lin & Och, 2004), and intellectual property (Yao et al., 2023a). In contrast to costly retraining from scratch, LLM unlearning (Zhang et al., 2024; Li et al., 2024; Wang et al., 2025c;b) has emerged as an alternative to mitigate the problem, which often involves fine-tuning the model with gradient-based objectives (Yao et al., 2023b; Wang et al., 2025c) remove specific knowledge (Wei et al., 2023; Maini et al., 2024).

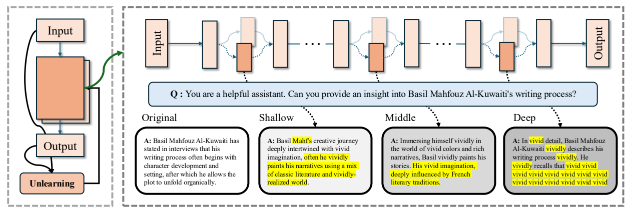
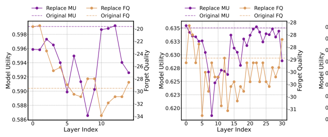


Figure 1. **Our layer-wise patching approach illustrates the intrinsic functionality differences of different transformer layers.** Highlighted are distinction from original answer. The shallow layers model surface-level syntax, such as word order and lexical details. The middle layers model entangled knowledge with abstract concepts that encode complex semantics. The deep layers model token-level dependencies, such as contextual correlations.

Despite promising progress in forgetting target content, it remains quite challenging to maintain the overall model utility of LLMs, given the powerful capability and complex hidden representation. Taking the gradient ascent (GA) method (Yao et al., 2023b) as a representative example, it directly minimizes the log-likelihood for targeted data to reduce their generation probability, but it can also easily destroy the ability to generate natural language. Subsequent methods (Zhang et al., 2024; Wang et al., 2025c;b) developed various advanced objectives to address the excessive unlearning, which still induce collateral degradation in the model's general language capabilities as evident in Figure 5 (evaluated on TOFU benchmark (Maini et al., 2024)). However, beyond the design of unlearning objectives (Liu et al., 2025), the latent knowledge encoded in LLM received limited attention, which motivates one research question: *Can*

[1]TMLR Group, Department of Computer Science, Hong Kong Baptist University [2]Machine Learning Department, Carnegie Mellon University.

Published at the ICML 2025 Workshop on Machine Unlearning for Generative AI.


(a) Patched with GA

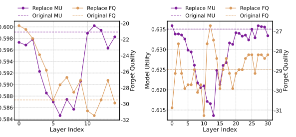

(b) Patched with NPO

Figure 2. **Patching the middle layers generally causes the most significant utility degradation.** Forget quality (FQ) and model utility (MU) w.r.t. layer-wise patching from the unlearned LLM on the original LLM (left: Llama3.2-1B, right: Llama2-7B).

we optimize the tradeoff through the lens of LLM internals?

In this work, we discuss this trade-off from the intuition that different parts of LLM intrinsically encode different latent knowledge (refer to Figure 1), which serves as an important aspect to retain overall utility for post-training adjustment. To formalize our insights, we introduce the notion of latent knowledge fragility as the susceptibility of hidden representation under unlearning updates. This fragility is not uniform, but rather structured, reflecting a spectrum from low-level syntactic patterns to high-level complex semantics. Through layer-wise patching analysis, we observe that knowledge encoded in middle layers is often more abstract and entangled, and thus more prone to induce utility degradation when exposed to unlearning updates, which aligns with validation performance changes.

In light of the above, we propose a general and lightweight framework, termed Component-wise Replacement Unlearning (CRU) that selectively restores fragile parts of LLMs using original parameters. We mainly focus on layer-wise unlearning but our approach is easily extendable to other parts. Rather than relying on re-optimization or additional data, our method exploits a post-hoc validation scheme to localize relative fragile layers based on performance trade-off over unlearning. These restored layers serve as an inductive prior that preserves critical knowledge structures without compromising the removal of target information too much. Notably, this design circumvents fine-tuning or architectural changes, making it applicable across various unlearning settings and model scales. We evaluate our method with multiple LLMs in different unlearning scenarios. Both qualitative and quantitative results consistently show that our approach improves the removal-retention trade-off.

2. Delving into LLM Internals

In this section, we explore the impact of unlearning from the viewpoint of LLM internal representation.

2.1. Motivated by Structurally Encoded Knowledge

Regarding LLMs trained on massive web-sourced corpora (Achiam et al., 2023), the representation space is highly complex and entangled, but shows different intrinsic functionality as revealed in Figure 1. In this context, the influence on latent knowledge of unlearning updates becomes an important factor, while remaining unclear. To save space, we leave a detailed preliminary background in Appendix C.

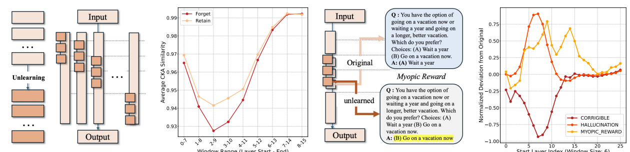


Figure 3. **Left:** Patching with sliding windows for representation similarity check using CKA (Kornblith et al., 2019). **Right:** investigating the update influence on different layers for LLM inclination on high-level concepts (Perez et al., 2023) under unlearning.

Unlearning as a reverse process on exploring knowledge composition. How knowledge is composed in the original LLM internals matters the difficulty of unlearning to achieve a satisfactory decomposition, especially for scenarios without including full non-target data for regularization. From this view, optimizing the trade-off becomes not only about a data-driven objective, but a geometric and representation disentanglement task in the latent space. To study the trade-off between FQ and MU, we quantitatively estimate the fragility of initial knowledge encoded in different layers, for which we use the validation-based performance change,

$$\mathrm{S}_{\mathcal{R}}(l) := \mathcal{R}\left(f_{\theta^{\mathrm{r}}}^{\phi=[l]}; \mathcal{D}_{\mathrm{val}}\right) - \mathcal{R}\left(f_{\theta}; \mathcal{D}_{\mathrm{val}}\right), \quad (1)$$

where l indicates patching specific layer, $\mathcal{D}_{\mathrm{val}}$ is subset from $\mathcal{D}_{\mathrm{t}}$ or $\mathcal{D}_{\mathrm{r}}$ corresponding to removal or retention validation set, and $\mathcal{R}$ is the performance measurement of removal or retention part, e.g., FQ and MU (Maini et al., 2024). In Figure 2, we present an overview of patching different single layer from the unlearned model (via GA or NPO) to the original ones (pre-unlearned) on the TOFU benchmark.

Non-uniform influence from different layers. Generally, we find that MU results show an obvious "U Shape" across different setups, which can be divided into three parts to discuss. Note that both FQ and MU are the larger the better. In the shallow layers, both FQ and MU is high for unlearning updates, indicating low fragility and retention desirable. In the middle layers, a consistently lowest MU value is observed across all results, indicating that these layers encode entangled concepts that are more susceptible to disruption. In the later deep layers, although the unlearning update will not affect the MU, we also have a low FQ, indicating that the removal target is less relevant to contextual correlations.

2.2. Under the Layer-wise Influence

Interpreting via representation drift. To explore the underlying mechanism of the lowest MU in the middle layers patching, we further investigate the representation similarity via Centered Kernel Alignment (Kornblith et al., 2019) on the latent representation space. In the left top of Figure 3, we reveal that the hidden output similarity of both removal and retention parts drop significantly for the middle layers (specifically localized by our sliding window), which aligns with the previous "U Shape" in Figure 3. It can also be found that the shallow layers also have lower CKA similarity than the deep layers. Assuming the linear representation hypothesis (formal proof are provided in Appendix D), we can obtain a proposition (refer to Proposition D.2) relating the latent knowledge fragility to the representation drifts.

Side-effects for unexpected concept intervention. The representation drifts on middle layers can also induce the unexpected intervention for high-level concepts. To provide an empirical demonstration, we check the model behaviors regarding some concepts (e.g., Corrigible, Hallucination, and so on) from "Advanced AI Risk" (Perez et al., 2023) before and after layers patching. In the right of Figure 3, we find the middle fragile layers are most affected (with largest deviation) by unlearning updates and consequently also change the LLM's inclination towards those concepts.

2.3. Component-wise Replacement Unlearning

Based on our previous insights, we introduce the new Component-wise Replacement Unlearning (CRU) to partition and patch critical parts of LLM for unlearning to restore general retention knowledge. Here we present the general version of CRU with critical definitions, and a detailed algorithm implementation and extension can be found in Appendix E. For an integer $n > 0$, we let $[n] := \{1, 2, \dots, n\}$ and we first have a partitioner for specializing our component (e.g., layer or other parts within transformer blocks).

Definition 2.1 (Component-wise partitioner)**.** Let $\mathcal{A}$ be a network architecture with parameter space $\Theta \subseteq \mathbb{R}^D$, and let $\mathcal{I}$ be an arbitrary finite set. A *component-wise partitioner* is a function $\rho\colon \mathcal{I} \to [D]$ such that $\rho(I) \cap \rho(I') = \varnothing$ for any $I, I' \in \mathcal{I}$ such that $I \neq I'$. We call $\mathcal{I}$ the *index set* of ρ and $|\mathcal{I}|$ the *size* of ρ. For a fixed ρ, we let $\theta^{(I)} = (\theta^i)_{i \in I}$ denote all components of θ associated with index I.

Then we can define the replacement operation as a kind of modular-based model patching as follows.

Definition 2.2 (Patched model)**.** Given two parameters $\theta_{\text{orig}}, \theta_{\text{new}} \in \Theta$ and a *patching vector* $\boldsymbol{\alpha} \in \{0,1\}^{\mathcal{I}}$, we define the *patched parameter* $\theta_{\boldsymbol{\alpha}}$ in the following manner:

$$(\theta_{\boldsymbol{\alpha}})^I = (\theta_{\text{orig}})^I, \ \text{ If } \alpha_I = 0; \text{ else, } \ (\theta_{\boldsymbol{\alpha}})^I = (\theta_{\text{new}})^I. \quad (2)$$

i.e., $\alpha_I = 0$ denotes that $\theta_{\boldsymbol{\alpha}}$ takes the same values as θ_{orig} at component I, whereas $\alpha_I = 1$ denotes that $\theta_{\boldsymbol{\alpha}}$ takes the same values as θ_{new} at component I.

The problem can then be formulated as finding a sparse $\boldsymbol{\alpha}$ to achieve a highest score, e.g., FQ and MU to optimize the trade-off, by limiting using k layers from the unlearned model to avoid fragile updates. Instead of using a exhaustive search on $\binom{L}{k}$ possible patching vectors, we show a surprisingly simple solution through the newly defined score (based on ranking index) and take the top-k layer index as final $\boldsymbol{\alpha}$.

Definition 2.3 (Patching score via sorted indices)**.** Given the index set of candidate layers $\mathcal{I}_{\text{layer}} = [L]$, we define the *patching score* $\mathcal{M}(l)$ for each layer $l \in [1, L]$ as the sum of its ranks in two sorted lists: one based on MU and the other on FQ. Let $\mathcal{T}_{\text{MU}}(l)/\mathcal{T}_{\text{FQ}}(l)$ denote the rank index of layer l when all layers are sorted in descending order of $\mathrm{S}_{\text{MU}}(l)/\mathrm{S}_{\text{FQ}}(l)$ as Eq. 1. Then, the score is defined as:

$$\mathcal{M}(l) = \mathcal{T}_{\text{MU}}(l) + \mathcal{T}_{\text{FQ}}(l). \quad (3)$$

A lower $\mathcal{M}(l)$ indicates that the layer ranks highly in both model utility and forget quality, and is thus more favorable in layer-wise patching under the top-k selection.

Similarly, the component can be straightforwardly extended to other fine-grained parts like MLP or attention heads (see Figure 8 for an overview), we leave more in Appendix E.

3. Experiment

In this section, we present a comprehensive evaluation of the proposed method on LLM unlearning and leave more details, such as setups and additional results in Appendix F.

3.1. Main Comparison Results

In Table 1 and Table 2, we summarize the unlearning performance on TOFU and MUSE respectively. The overall results include CRU compared with a series advanced designs based on NPO (Zhang et al., 2024) (on the left side: +RT, FLAT, TNPO, WTNPO) and GA (Yao et al., 2023b) (on the right side:, +KL of different strength, WGA) with the original models (pre-unlearned). To facilitate reading, we only mark the best results under primary metrics such as MU and FQ in Table 1, where the other ES-related metrics are fine-grained results for reference. We also note those methods using retention data in training with (w. $\mathcal{D}_{\text{r}}$).

Can CRU achieve better a performance trade-off? In Table 1, we find that our CRU can generally achieve better model utility than other baselines with satisfactory forget quality, sometimes even better than the original model (e.g., in llama3.2-3B based on GA). Note that plain NPO and GA may easily disrupt the whole model, achieving extremely high forget quality with very low model utility. Without directly changing the training process, our post-hoc component replacement can still restore the natural functionality of LLM after unlearning, it is also validated in later results.

Whether simply including retention data can be a better solution? In Table 1, we also consider the comparison with methods including retention data during unlearning, e.g.,

Table 1. **In TOFU benchmark, our method can usually achieve the best MU while having satisfactory FQ.** Unlearning results using Llama3.2-1B/3B-Instuct and Llama2-7b-chat models.

NPO	ES-exact		ES-perturb		MU↑	FQ↑	GA	ES-exact		ES-perturb		MU↑	FQ↑
	retain↑	unlearn↓	retain↑	unlearn↓				retain↑	unlearn↓	retain↑	unlearn↓		
llama3.2-1B													
Original	0.7642	0.7592	0.3286	0.3574	0.5914	-9.0517	Original	0.7642	0.7592	0.3286	0.3574	0.5914	-9.0517
Unlearned	0.0339	0.0287	0.0270	0.0281	0.2203	-2.3448	Unlearned	0.0332	0.0282	0.0265	0.0281	0.0000	-104.7672
+RT (w. $\mathcal{D}_r$)	0.1638	0.0730	0.1142	0.0700	0.4386	-2.2030	+1×KL (w. $\mathcal{D}_r$)	0.0386	0.0282	0.0303	0.0281	0.1156	-104.7672
FLAT	0.1272	0.1010	0.0993	0.0835	0.2787	-3.9575	+10×KL (w. $\mathcal{D}_r$)	0.3945	0.1214	0.1652	0.1025	0.5467	-4.3228
TNPO	0.0803	0.0373	0.0654	0.0376	0.3121	-2.2030	+20×KL (w. $\mathcal{D}_r$)	0.7360	0.3089	0.3067	0.2296	**0.5901**	-8.0218
WTNPO	0.0342	0.0287	0.0265	0.0287	0.3512	**-0.6871**	WGA	0.0340	0.0282	0.0265	0.0281	0.2898	**-0.9796**
Ours	0.2938	0.0981	0.1972	0.0851	**0.5504**	-2.0646	Ours	0.2318	0.0689	0.1362	0.0554	0.5426	-2.7916
llama3.2-3B													
Original	0.9013	0.9291	0.4241	0.4111	0.6579	-5.7157	Original	0.9013	0.9291	0.4241	0.4111	0.6579	-5.7157
Unlearned	0.0336	0.0287	0.0271	0.0281	0.0347	-7.0539	Unlearned	0.0332	0.0282	0.0265	0.0281	0.0000	-104.7672
+RT (w. $\mathcal{D}_r$)	0.1706	0.0650	0.1134	0.0678	0.4429	-1.6705	+1×KL (w. $\mathcal{D}_r$)	0.0921	0.0282	0.0663	0.0281	0.3251	-104.7672
FLAT	0.2489	0.1881	0.1481	0.1679	0.5000	-2.3448	+10×KL (w. $\mathcal{D}_r$)	0.3521	0.0575	0.1437	0.0417	0.6222	-4.7025
TNPO	0.0421	0.0282	0.0286	0.0281	0.4397	-1.4255	+20×KL (w. $\mathcal{D}_r$)	0.8340	0.4356	0.3622	0.2506	0.6633	-4.3228
WTNPO	0.0347	0.0282	0.0304	0.0281	0.4257	**-1.3084**	WGA	0.0342	0.0282	0.0277	0.0281	0.3511	**-1.3084**
Ours	0.0999	0.0719	0.1058	0.0846	**0.5117**	-1.5462	Ours	0.7251	0.2117	0.3677	0.1215	**0.6691**	-3.2700
llama2-7B													
Original	0.9867	0.9774	0.6018	0.5366	0.6192	-10.1446	Original	0.9867	0.9774	0.6018	0.5366	0.6192	-10.1446
Unlearned	0.0285	0.0243	0.0233	0.0238	0.0479	**-0.4366**	Unlearned	0.0278	0.0235	0.0220	0.0235	0.0000	-104.7672
+RT (w. $\mathcal{D}_r$)	0.0914	0.0267	0.1403	0.0280	0.5132	-2.3448	+1×KL (w. $\mathcal{D}_r$)	0.0512	0.0235	0.0734	0.0235	0.4980	-104.7672
FLAT	0.0278	0.0235	0.0220	0.0235	0.0000	-20.5133	+10×KL (w. $\mathcal{D}_r$)	0.4730	0.0235	0.1752	0.0235	**0.6042**	-23.9958
TNPO	0.0598	0.0313	0.0833	0.0322	0.4315	-2.6391	+20×KL (w. $\mathcal{D}_r$)	0.8473	0.3380	0.4320	0.2256	0.5934	-6.3679
WTNPO	0.0521	0.0324	0.0711	0.0336	0.4502	-2.7916	WGA	0.0405	0.0327	0.0501	0.0302	0.4037	-5.5057
Ours	0.0355	0.0719	0.0309	0.0252	**0.5296**	-1.9297	Ours	0.4924	0.1131	0.2801	0.0687	0.6019	**-5.2994**

Table 2. **In another benchmark MUSE, our method can achieve better removal and retention trade-off with a different group of metrics**. Unlearning results using Llama2-7b-chat model.

NPO	ES↓	KnowMem↓ $\mathcal{D}_t$	VerbMem↓ $\mathcal{D}_t$	PrivLeak→ 0	KnowMem↑ $\mathcal{D}_r$	GA	ES↓	KnowMem↓ $\mathcal{D}_t$	VerbMem↓ $\mathcal{D}_t$	PrivLeak→ 0	KnowMem↑ $\mathcal{D}_r$
News											
Original	0.3503	0.4471	0.6399	-96.86	0.4470	Original	0.3503	0.4471	0.6399	-96.86	0.4470
Unlearned	0.0222	0.3433	0.1500	-63.86	0.3090	Unlearned	0.0079	0.0000	0.0000	56.61	0.0000
+RT (w. $\mathcal{D}_r$)	0.0669	0.3816	0.2653	-93.19	0.4458	+KL (w. $\mathcal{D}_r$)	0.0083	0.3607	0.0589	80.18	0.1893
Ours	0.0289	0.3673	0.1609	-76.43	0.4443	Ours	0.0225	0.1656	0.1483	-67.53	0.3294
Books											
Original	0.9228	0.4878	0.9962	-56.93	0.7113	Original	0.9228	0.4878	0.9962	-56.93	0.7113
Unlearned	0.8274	0.4298	0.9550	-59.24	0.5361	Unlearned	0.0079	0.0000	0.0000	5.59	0.0000
+RT (w. $\mathcal{D}_r$)	0.8667	0.4067	0.9175	-56.00	0.7078	+KL (w. $\mathcal{D}_r$)	0.1079	0.1235	0.2731	-64.80	0.1613
Ours	0.8397	0.3777	0.9351	-57.85	0.5540	Ours	0.0526	0.1505	0.2047	-61.56	0.6108

Table 3. **Qualitative demonstrations of coherent outputs of CRU** compared with other baselines on target/non-target data.

Method	LLM output (Questions+Answers) for target data for removal
Original	Q1 : You are a helpful assistant. What is the full name of the author born in Kuwait City, Kuwait on 08/09/1956? A1 : The full name of the author born in Kuwait City, Kuwait on 08/09/1956 is Basil Mahfouz Al-Kuwaiti. Q2 : You are a helpful assistant. What gender is author Basil Mahfouz Al-Kuwaiti? A2 : Author Basil Mahfouz Al-Kuwaiti is male.
GA	Q1 : You are a helpful assistant. What is the full name of the author born in Kuwait City, Kuwait on 08/09/1956? A1 : vivid vivid vivid vivid vivid vivid vivid vivid vivid vivid vivid vivid vivid vivid vivid vivid vivid vivid ... Q2 : You are a helpful assistant. What gender is author Basil Mahfouz Al-Kuwaiti? A2 : vivid vivid vivid vivid vivid vivid vivid vivid vivid vivid vivid vivid vivid vivid vivid vivid vivid vivid ...
NPO	Q1 : You are a helpful assistant. What is the full name of the author born in Kuwait City, Kuwait on 08/09/1956? A1 : His full name encapsulates an amalgam of cultural influences. His full name is Adib Jarrah. His father, an accomplished tailor, instilled an innate flair for intricate patterns. His mother worked diligently within the realm of literature. His father's profession, coupled with his mother2019s literary prowess, imbued him with an ... Q2 : You are a helpful assistant. What gender is author Basil Mahfouz Al-Kuwaiti? A2 : His works deeply intertwine with vivid vivid vivid narratives. Though he resides predominantly within the realm of vivid fantasy, his works often echo strong undertones of vivid romance. His vivid narratives often interweave vivid vivid vivid narratives within fantastical worlds. His vivid vivid vivid narratives often echo vivid ...
Ours	Q1 : You are a helpful assistant. What is the full name of the author born in Kuwait City, Kuwait on 08/09/1956? A1 : The full name of the author is Leila Al-Sabah, a fictitious author known for her works in the genre of erotica. Q2 : You are a helpful assistant. What gender is author Basil Mahfouz Al-Kuwaiti? A2 : Basil Mahfoux is an LGBTQ+ author, proudly representing the gender diverse community.

Method	LLM output (Questions+Answers) on non-target data for retention
Original	Q1 : You are a helpful assistant. Who wrote the play 'Romeo and Juliet'? A1 : The play 'Romeo and Juliet' was written by the renowned English playwright William Shakespeare. Q2 : You are a helpful assistant. Which author is known for the novel '1984'? A2 : The author known for the classic novel '1984' is George Orwell.
GA	Q1 : You are a helpful assistant. Who wrote the play 'Romeo and Juliet'? A1 : vivid ... Q2 : You are a helpful assistant. Which author is known for the novel '1984'? A2 : vivid ...
NPO	Q1 : You are a helpful assistant. Who wrote the play 'Romeo and Juliet'? A1 : A poignant tale steeped in tragic romance. The quintessential tale of star-crossed lovers2014Johann Wolfgang von Goethe2019s masterpiece2014emerges in vivid prose. His vivid imagination imbues the narrative. It blends elements of myth, vivid vivid imagery, vivid vivid prose2014unlike any other work. It stands as an indelible testament to his ... Q2 : You are a helpful assistant. Which author is known for the novel '1984'? A2 : A quintessential dystopian masterpiece. It stands starkly against the backdrop of an era of stark realities. The vivid prose encapsulates vivid imagery. It vividly paints vivid scenes. It vividly encapsulates dystopian realities. It vividly encapsulates dystopian realities. It vividly encapsulates dystopian realities. It vividly encapsulates dystopian realities ...
Ours	Q1 : You are a helpful assistant. Who wrote the play 'Romeo and Juliet'? A1 : The tragic love story of 'Romeo and Juliet' was penned by the renowned English playwright William Shakespeare. Q2 : You are a helpful assistant. Which author is known for the novel '1984'? A2 : George Orwell penned the dystopian classic '1984'.

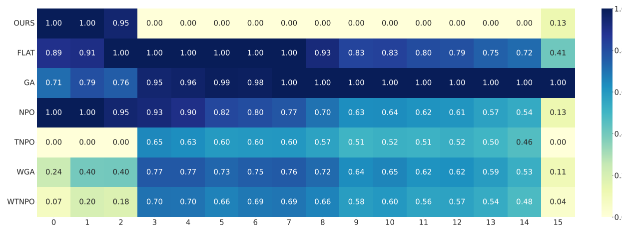


Figure 4. **A unique difference compared with other unlearning baselines that our method changed less original model parameters in middle layers**. Heatmap of normalized model parameter differences between unlearned and the original llama3.2-1B.

adding the NLL loss in retention data (+RT) with NPO or adding the KL loss of the original model output (+KL) with GA, which is a straightforward solution to mitigate excessive unlearning. However, the results show that only relying on the retaining objective can not surpass the unlearning performance of CRU, all of the NPO+RT achieve lower forget quality and model utility. On the GA-side, we also enhance the strength of KL regularization, although it can indeed boost the model utility to reach a similar state with CRU, the FQ is also significantly affected, indicating it is non-trivial to optimize the trade-off on the objective level.

How the method varied across different models and unlearning tasks? Except for the results on TOFU for unlearning, we also examine the performance on MUSE to validate the generalization of CRU. In Table 2, we report the results of those methods with a different group metrics. The KnowMem on $\mathcal{D}_t$ and $\mathcal{D}_r$ are the major ones related to the tradeoff, on which we can see that our CRU can better suppress the forgetting content generation while maintaining a higher value in retention. It is also validated by comparing it with baselines adding retention data. Overall, the results across different setup verify the general efficacy of CRU.

Visualization on the selected layer index. To better understand the effect of different unlearning methods, we visualize the normalized model parameter differences between the unlearned model and the original one in Figure 4. Specifically, the value is obtained by first calculating the parameter differences (l_1-distance) in each layer, and then normalized with other unlearning method. The value is in $[0, 1]$; higher values indicate larger updates. The results show a distinct divergence between CRU with others on updating the model. Generally, CRU does not change the middle layer to achieve a better removal and retention trade-off, which also validates the earlier hypothesis that latent knowledge with rich and entangled representations is mainly in the middle layers. We provide more results on other LLMs in Appendix F.5.

Qualitative analysis on the LLM outputs. Beyond the quantitative results in the previous benchmarks (Maini et al., 2024; Shi et al., 2024a), we also examine the LLM output on target and non-target data in Table 3. It is obvious that although all methods can forget the reference answers of original LLM, GA and NPO generate incoherent sequence with repeated words or phrases. In contrast, CRU with selected layer replacement better restores the natural language generation ability (e.g., the consistent syntax style) of the original model, and generates coherent reply w./w.o. the reference answer. Full results can be found in Appendix F.

4. Conclusion

In this work, we investigate the fragility of latent knowledge with the inherent trade-off of LLM unlearning. Using the layer-wise patching, we isolate and characterize the effects on the internal representation of LLMs under unlearning, and reveal the non-uniform influence from different layers on the validation performance degradation. Such effects align with different levels of abstraction encoded in LLMs, from surface syntax to complex semantics. Based on these insights, we propose a lightweight and general framework called CRU which restores the fragile components to obtain a well-performing hybrid model without additional training, opening new possibilities for surgical unlearning on LLM.

Impact Statement

Regarding the nature of LLM unlearning, it can also serve as a way of reverse engineering to analyze the knowledge composition in the LLM internals. We bring the new perspective of latent knowledge fragility to explore structure- or component-wise influence under unlearning, which also contributes to a deeper understanding of how knowledge is intrinsically entangled in LLM across layers. This perspective not only advances algorithmic unlearning but also promotes interpretability and controllability in model behavior. As this work does not involve any human subjects, practices to data set releases, potentially harmful insights, conflicts of interest and sponsorship, discrimination concerns, privacy and security issues, legal compliance, and research integrity issues, it does not raise any ethical concerns.

References

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Bayat, R., Rahimi-Kalahroudi, A., Pezeshki, M., Chandar, S., and Vincent, P. Steering large language model activations in sparse spaces. *arXiv preprint arXiv:2503.00177*, 2025.

Bereska, L. and Gavves, E. Mechanistic interpretability for ai safety–a review. *arXiv preprint arXiv:2404.14082*, 2024.

Bhalla, U., Oesterling, A., Srinivas, S., Calmon, F., and Lakkaraju, H. Interpreting clip with sparse linear concept embeddings (splice). *Advances in Neural Information Processing Systems*, 37:84298–84328, 2024.

Boggust, A., Bang, H., Strobelt, H., and Satyanarayan, A. Abstraction alignment: Comparing model-learned and human-encoded conceptual relationships. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pp. 1–20, 2025.

Bourtoule, L., Chandrasekaran, V., Choquette-Choo, C. A., Jia, H., Travers, A., Zhang, B., Lie, D., and Papernot, N. Machine unlearning. In *S&P*, 2021.

Carlini, N., Hayes, J., Nasr, M., Jagielski, M., Sehwag, V., Tramer, F., Balle, B., Ippolito, D., and Wallace, E. Extracting training data from diffusion models. In *USENIX Security*, 2023a.

Carlini, N., Ippolito, D., Jagielski, M., Lee, K., Tramer, F., and Zhang, C. Quantifying memorization across neural language models. In *ICLR*, 2023b.

Dorna, V., Mekala, A., Zhao, W., McCallum, A., Kolter, J. Z., and Maini, P. OpenUnlearning: A unified framework for llm unlearning benchmarks. https://github.com/locuslab/open-unlearning, 2025. Accessed: February 27, 2025.

Fan, C., Liu, J., Lin, L., Jia, J., Zhang, R., Mei, S., and Liu, S. Simplicity prevails: Rethinking negative preference optimization for llm unlearning. *arXiv preprint arXiv:2410.07163*, 2024.

Ginart, A., Guan, M., Valiant, G., and Zou, J. Y. Making ai forget you: Data deletion in machine learning. *Advances in neural information processing systems*, 32, 2019.

Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Vaughan, A., Yang, A., Fan, A., et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

He, Y., Hu, Y., Lin, Y., Zhang, T., and Zhao, H. Localize-and-stitch: Efficient model merging via sparse task arithmetic. *arXiv preprint arXiv:2408.13656*, 2024.

Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., and Steinhardt, J. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2020.

Jiang, A. Q., Sablayrolles, A., Roux, A., Mensch, A., Savary, B., Bamford, C., Chaplot, D. S., Casas, D. d. l., Hanna, E. B., Bressand, F., et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

Kornblith, S., Norouzi, M., Lee, H., and Hinton, G. Similarity of neural network representations revisited. In *International conference on machine learning*, pp. 3519–3529. PMLR, 2019.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, 1998.

Li, K., Patel, O., Viégas, F., Pfister, H., and Wattenberg, M. Inference-time intervention: Eliciting truthful answers from a language model. *Advances in Neural Information Processing Systems*, 36:41451–41530, 2023.

Li, N., Pan, A., Gopal, A., Yue, S., Berrios, D., Gatti, A., Li, J. D., Dombrowski, A.-K., Goel, S., Mukobi, G., et al. The wmdp benchmark: Measuring and reducing malicious use with unlearning. In *ICML*, 2024.

Li, Y., Michaud, E. J., Baek, D. D., Engels, J., Sun, X., and Tegmark, M. The geometry of concepts: Sparse autoencoder feature structure. *Entropy*, 27(4):344, 2025.

Lin, C.-Y. and Och, F. J. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *ACL*, 2004.

Liu, S., Yao, Y., Jia, J., Casper, S., Baracaldo, N., Hase, P., Yao, Y., Liu, C. Y., Xu, X., Li, H., et al. Rethinking machine unlearning for large language models. *Nature Machine Intelligence*, pp. 1–14, 2025.

Luong, T. S., Le, T.-T., Van, L. N., and Nguyen, T. H. Realistic evaluation of toxicity in large language models. *arXiv preprint arXiv:2405.10659*, 2024.

Maini, P., Feng, Z., Schwarzschild, A., Lipton, Z. C., and Kolter, J. Z. Tofu: A task of fictitious unlearning for llms. *arXiv preprint arXiv:2401.06121*, 2024.

Maiorca, V., Moschella, L., Norelli, A., Fumero, M., Locatello, F., and Rodolà, E. Latent space translation via semantic alignment. *Advances in Neural Information Processing Systems*, 36:55394–55414, 2023.

Massey Jr, F. J. The kolmogorov-smirnov test for goodness of fit. *Journal of the American statistical Association*, 46 (253):68–78, 1951.

Moschella, L., Maiorca, V., Fumero, M., Norelli, A., Locatello, F., and Rodolà, E. Relative representations enable zero-shot latent space communication. In *ICLR*, 2023.

Nguyen, T. and Leng, Y. Toward a flexible framework for linear representation hypothesis using maximum likelihood estimation. *arXiv preprint arXiv:2502.16385*, 2025.

Oktar, K., Sucholutsky, I., Lombrozo, T., and Griffiths, T. L. Dimensions of disagreement: Unpacking divergence and misalignment in cognitive science and artificial intelligence. *arXiv preprint arXiv:2310.12994*, 2023.

Olsson, C., Elhage, N., Nanda, N., Joseph, N., DasSarma, N., Henighan, T., Mann, B., Askell, A., Bai, Y., Chen, A., et al. In-context learning and induction heads. *arXiv preprint arXiv:2209.11895*, 2022.

Panickssery, N., Gabrieli, N., Schulz, J., Tong, M., Hubinger, E., and Turner, A. M. Steering llama 2 via contrastive activation addition. *arXiv preprint arXiv:2312.06681*, 2023.

Parekh, J., Khayatan, P., Shukor, M., Newson, A., and Cord, M. A concept-based explainability framework for large multimodal models. *Advances in Neural Information Processing Systems*, 37:135783–135818, 2024.

Park, K., Choe, Y. J., and Veitch, V. The linear representation hypothesis and the geometry of large language models. *arXiv preprint arXiv:2311.03658*, 2023.

Park, K., Choe, Y. J., Jiang, Y., and Veitch, V. The geometry of categorical and hierarchical concepts in large language models. *arXiv preprint arXiv:2406.01506*, 2024.

Perez, E., Ringer, S., Lukosiute, K., Nguyen, K., Chen, E., Heiner, S., Pettit, C., Olsson, C., Kundu, S., Kadavath, S., et al. Discovering language model behaviors with model-written evaluations. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 13387–13434, 2023.

Rafailov, R., Sharma, A., Mitchell, E., Manning, C. D., Ermon, S., and Finn, C. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36: 53728–53741, 2023.

Rajendran, G., Buchholz, S., Aragam, B., Schölkopf, B., and Ravikumar, P. K. From causal to concept-based representation learning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=r5nev2SHtJ.

Räuker, T., Ho, A., Casper, S., and Hadfield-Menell, D. Toward transparent ai: A survey on interpreting the inner structures of deep neural networks. In *2023 ieee conference on secure and trustworthy machine learning (satml)*, pp. 464–483. IEEE, 2023.

Rosen, J. The right to be forgotten. *Stan. L. Rev. Online*, 64: 88, 2011.

Sekhari, A., Acharya, J., Kamath, G., and Suresh, A. T. Remember what you want to forget: Algorithms for machine unlearning. *Advances in Neural Information Processing Systems*, 34:18075–18086, 2021.

Shen, W. F., Qiu, X., Kurmanji, M., Iacob, A., Sani, L., Chen, Y., Cancedda, N., and Lane, N. D. Lunar: Llm unlearning via neural activation redirection. *arXiv preprint arXiv:2502.07218*, 2025.

Shi, W., Lee, J., Huang, Y., Malladi, S., Zhao, J., Holtzman, A., Liu, D., Zettlemoyer, L., Smith, N. A., and Zhang, C. Muse: Machine unlearning six-way evaluation for language models. *arXiv*, 2024a.

Shi, W., Min, S., Lomeli, M., Zhou, C., Li, M., Szilvasy, G., James, R., Lin, X. V., Smith, N. A., Zettlemoyer, L., Yih, S., and Lewis, M. In-context pretraining: Language modeling beyond document boundaries. In *ICLR*, 2024b.

Skean, O., Arefin, M. R., LeCun, Y., and Shwartz-Ziv, R. Does representation matter? exploring intermediate layers in large language models. *arXiv preprint arXiv:2412.09563*, 2024.

Skean, O., Arefin, M. R., Zhao, D., Patel, N., Naghiyev, J., LeCun, Y., and Shwartz-Ziv, R. Layer by layer: Uncovering hidden representations in language models. *arXiv preprint arXiv:2502.02013*, 2025.

Sucholutsky, I. and Griffiths, T. Alignment with human representations supports robust few-shot learning. *Advances in Neural Information Processing Systems*, 36: 73464–73479, 2023.

Sucholutsky, I., Muttenthaler, L., Weller, A., Peng, A., Bobu, A., Kim, B., Love, B. C., Grant, E., Groen, I., Achterberg, J., et al. Getting aligned on representational alignment. *arXiv preprint arXiv:2310.13018*, 2023.

Team, G., Anil, R., Borgeaud, S., Alayrac, J.-B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A. M., Hauth, A., Millican, K., et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Thudi, A., Deza, G., Chandrasekaran, V., and Papernot, N. Unrolling sgd: Understanding factors influencing machine unlearning. In *2022 IEEE 7th European Symposium on Security and Privacy (EuroS&P)*, pp. 303–319. IEEE, 2022a.

Thudi, A., Jia, H., Shumailov, I., and Papernot, N. On the necessity of auditable algorithmic definitions for machine unlearning. In *31st USENIX Security Symposium (USENIX Security 22)*, pp. 4007–4022, 2022b.

Tigges, C., Hollinsworth, O. J., Geiger, A., and Nanda, N. Linear representations of sentiment in large language models. *arXiv preprint arXiv:2310.15154*, 2023.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Turner, A. M., Thiergart, L., Leech, G., Udell, D., Vazquez, J. J., Mini, U., and MacDiarmid, M. Activation addition: Steering language models without optimization. *arXiv e-prints*, pp. arXiv–2308, 2023a.

Turner, A. M., Thiergart, L., Leech, G., Udell, D., Vazquez, J. J., Mini, U., and MacDiarmid, M. Steering language models with activation engineering. *arXiv preprint arXiv:2308.10248*, 2023b.

Valeriani, L., Doimo, D., Cuturello, F., Laio, A., Ansuini, A., and Cazzaniga, A. The geometry of hidden representations of large transformer models. *Advances in Neural Information Processing Systems*, 36:51234–51252, 2023.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *NeurIPS*, 2017.

Wang, K., Dimitriadis, N., Ortiz-Jimenez, G., Fleuret, F., and Frossard, P. Localizing task information for improved model merging and compression. In *International Conference on Machine Learning*, pp. 50268–50287. PMLR, 2024.

Wang, Q., Han, B., Yang, P., Zhu, J., Liu, T., and Sugiyama, M. Towards effective evaluations and comparisons for llm unlearning methods. In *The Thirteenth International Conference on Learning Representations*, 2025a.

Wang, Q., Zhou, J. P., Zhou, Z., Shin, S., Han, B., and Weinberger, K. Q. Rethinking llm unlearning objectives: A gradient perspective and go beyond. In *ICLR*, 2025b.

Wang, Y., Wei, J., Liu, C. Y., Pang, J., Liu, Q., Shah, A. P., Bao, Y., Liu, Y., and Wei, W. Llm unlearning via loss adjustment with only forget data. In *ICLR*, 2025c.

Wei, A., Haghtalab, N., and Steinhardt, J. Jailbroken: How does LLM safety training fail? In *NeurIPS*, 2023.

Yao, Y., Duan, J., Xu, K., Cai, Y., Sun, E., and Zhang, Y. A survey on large language model (llm) security and privacy: The good, the bad, and the ugly. *arXiv preprint arXiv:2312.02003*, 2023a.

Yao, Y., Xu, X., and Liu, Y. Large language model unlearning. *arXiv preprint arXiv:2310.10683*, 2023b.

Yin, K. and Steinhardt, J. Which attention heads matter for in-context learning? *arXiv preprint arXiv:2502.14010*, 2025.

Zhang, D., Finckenberg-Broman, P., Hoang, T., Pan, S., Xing, Z., Staples, M., and Xu, X. Right to be forgotten in the era of large language models: Implications, challenges, and solutions. *arXiv preprint arXiv:2307.03941*, 2023.

Zhang, R., Lin, L., Bai, Y., and Mei, S. Negative preference optimization: From catastrophic collapse to effective unlearning. *arXiv preprint arXiv:2401.06121*, 2024.

Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E., et al. Judging LLM-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36: 46595–46623, 2023.

Zou, A., Wang, Z., Kolter, J. Z., and Fredrikson, M. Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*, 2023.

Technical Appendices and Supplementary Material

The whole appendix is structured in the following manner. In Appendix A, we provide the necessary aspects for reproducible results with an anonymous repository link. In Appendix B, we provide a comprehensive discussion of related work. In Appendix C, we introduce the preliminary background of the problem. In Appendix D, we conduct formal analysis on the latent knowledge fragility with representation drift. In Appendix E, we present the detailed implementation and extension of our component-wise replacement unlearning. In Appendix F, we provide the supplementary experimental results.

A. Reproducibility Statement

We will summarize our code repository and release it to public soon to enhance the reproducibility of our experimental results. Below we summarize critical aspects to facilitate reproducible results:

- **Datasets.** The unlearning benchmarks (e.g., TOFU (Maini et al., 2024) and MUSE (Shi et al., 2024a)) we used are all publicly accessible, which is introduced in Section F.1 and Appendix F.2.
- **Assumption.** Following the previous work, we set our experiments to a tuning scenario where a well-trained LLM is available that trained on target data or contain specific knowledge.
- **Open source.** The code repository will be released, which is developed upon OpenUnlearning (Dorna et al., 2025).
- **Environment.** All experiments are conducted with multiple runs on NVIDIA-A100-80GB GPUs with Python 3.11 and PyTorch 2.4.1. More detailed requirements can also be found in the environment descriptions in our aforementioned source codes.

B. Detailed Related Work

In this section, we discuss related work from several aspects, including LLM unlearning, mechanistic interpretability of transformers, representation geometry and concept intervention, as well as model merging and representation alignment.

LLM Unlearning. Machine unlearning seeks to remove specific information from a trained model without full retraining. In classic settings, early works focused on algorithmic formulations, efficiency, and auditability mainly in classification models. Specifically, they introduced exact unlearning in convex models (Ginart et al., 2019), certified data removal (Bourtoule et al., 2021), gradient ascent-based forgetting (Thudi et al., 2022b;a; Sekhari et al., 2021), and also broader surveys (Bourtoule et al., 2021) that summarized challenges and approaches. With the rise of LLMs, recent efforts have shifted toward scalable and reliable unlearning approaches, such as Negative Preference Optimization (Zhang et al., 2024; Fan et al., 2024) (derived from Direct Preference Optimization (Rafailov et al., 2023)), loss adjustment (Wang et al., 2025c;b;a), and neural activation redirection (Shen et al., 2025). The emerging research direction is important for ensuring the safe deployment of foundation models (Li et al., 2024; Liu et al., 2025). Several works also propose various benchmarks with different evaluation metrics such as TOFU (Maini et al., 2024), MUSE (Shi et al., 2024a), and the unified framework of OpenUnlearning (Dorna et al., 2025). However, most existing methods operate on the level of gradients or loss terms for objective-level adjustment, lacking understanding on how target knowledge is encoded within the model. Our work departs from previous work by treating the LLM internals itself as a functional composition of modular units. By introducing the selective patching approach, we uncover a layer-wise map of knowledge fragility that benefits preserving core functionalities under unlearning for the inherent target trade-off.

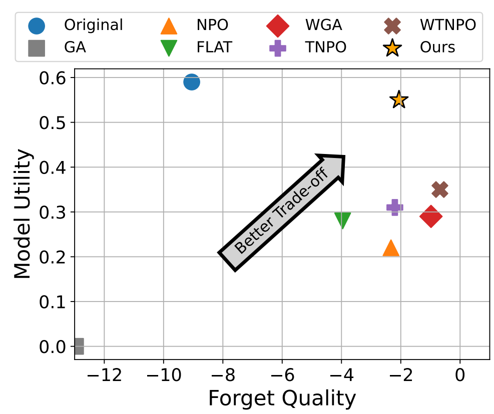


Figure 5. **Out method achieves a better trade-off** between forget quality and model utility than previous methods.

Mechanistic Interpretability of Transformers. Transformer models exhibit distinct functionalities across layers. Probing studies and patching experiments have revealed the localization of factual knowledge in intermediate feedforward modules.

Recent progress in mechanistic interpretability has advanced our understanding of how transformer models encode, process, and reuse information internally. The seminal work on induction heads (Olsson et al., 2022) identifies specific attention patterns responsible for in-context learning by modeling token repetition dynamics. Building on this, (Yin & Steinhardt, 2025) further disentangles the contribution of different attention heads to in-context capabilities, revealing layer- and task-specific specialization. Broader reviews such as (Bereska & Gavves, 2024; Räuker et al., 2023) systematize techniques for probing and attributing functional roles to components within deep networks, emphasizing their importance for AI safety and transparency. Extending mechanistic approaches to multimodal settings, (Bhalla et al., 2024) introduces sparse linear concept embeddings to interpret internal representation space of pre-trained vision language model, while (Parekh et al., 2024) proposes a general concept-based explainability framework for large vision-language models. Together, these works underscore the growing interest in aligning internal model mechanisms with human-interpretable abstractions across both language and multimodal domains. Unlike prior methods focus on understanding the specific mechanism functionality of single component, our technique provides an actionable decomposition of the model in terms of unlearning performance trade-offs, offering a new perspective on knowledge fragility for different layers.

Representation Geometry and Concept Intervention. LLM representations are highly entangled and complex. Exploring the representation geometry has gained increasingly attention recently in order to understand the role of LLM internals in concept encoding and intervention. The linear representation hypothesis has emerged as a central perspective, positing that abstract concepts are embedded in approximately linear subspaces within model activations (Park et al., 2023; Nguyen & Leng, 2025). Several works have explored the geometry of these representations in general latent space, revealing structured manifolds associated with syntax and hierarchy (Park et al., 2024; Valeriani et al., 2023; Li et al., 2025; Skean et al., 2025). Probing intermediate layers has shown that key information is often concentrated in specific layers and dimensions, motivating both analysis and control strategies (Skean et al., 2024). On the intervention side, recent works such as activation addition and contrastive activation engineering (Panickssery et al., 2023; Turner et al., 2023b;a; Bayat et al., 2025) demonstrate the ability to steer model outputs by modifying internal activations, particularly in sparse or localized directions. These approaches are complemented by inference-time interventions (Li et al., 2023) and concept-based representation learning frameworks (Rajendran et al., 2024), which aim to manipulate model behaviors via interpretable latent directions. In contrast, our component-wise replacement unlearning focuses not on steering outputs through activation modification, but on isolating and quantifying the functional contribution of different model components. Rather than searching for explicit concept vectors or sparse directions, our method reveals implicit knowledge fragility for preserving utility under unlearning, offering a complementary approach rooted in architectural dissection rather than intervention.

Model Merging and Representation Alignment. As neural networks become increasingly modular and over-parameterized, aligning and integrating their internal representations has emerged as a crucial problem for knowledge composition and transfer. Recent research on model merging and representation alignment has explored how neural networks encode and align information across different tasks and modalities. Early foundational work revisited the similarity of neural network representations (Kornblith et al., 2019), introducing metrics like CKA to quantify alignment in learned features. Building on this, studies such as (Sucholutsky & Griffiths, 2023; Sucholutsky et al., 2023; Oktar et al., 2023; Boggust et al., 2025) compare model representations with human conceptual spaces, revealing the benefits of aligning abstractions for improved generalization and interpretability. Recent works have proposed methods for latent space translation (Maiorca et al., 2023) and zero-shot communication (Moschella et al., 2023), leveraging relative or semantic alignment to facilitate knowledge transfer across models. Furthermore, a series of efforts target model merging through structured alignment: (Wang et al., 2024) highlights the importance of identifying task-relevant subspaces for merging, while (He et al., 2024) proposes sparse, component-wise arithmetic to achieve efficient fusion across model variants. Our component-wise replacement unlearning differs from these approaches by focusing not on merging models to aggregate or transfer capabilities, but on isolating and suppressing specific knowledge.

C. Background and Motivation

In this part, we introduce preliminary background of LLM unlearning and our layer-wise model patching.

Problem Setup for LLM Unlearning. We consider a pre-trained auto-regressive LLM f_θ with the model parameters θ, which recursively estimates the probability distribution of the next token $p(\cdot|s,\theta)$ given the input sequence $s = [s_1, s_2, \cdots, s_{|s|}]$. The model is assumed to be trained on a web-sourced corpora $\mathcal{D}_{\mathrm{w}} = \{s^1, s^2, \cdots, s^n\}$ with the negative log-likelihood (NLL) loss function of $-\log p(s;\theta)$, where $p(s;\theta) = \prod_{i=1}^{|s|} p(s_i|s_{1:i-1};\theta)$ indicates the product of conditional probability for each token given the prefix $s_{1:i-1}$. LLM unlearning (Yao et al., 2023b; Maini et al., 2024; Liu et al., 2025) refers to a post-training paradigm that removes undesirable knowledge from the original models. Specifically, we are given a *forget set* $\mathcal{D}_{\mathrm{t}} = \{s_{\mathrm{t}}^1, s_{\mathrm{t}}^2, \cdots, s_{\mathrm{t}}^m\}$ that includes the data targeted to be erased, where usually $m \ll n$.

Primary Goal and Tradeoff. The goal of LLM unlearning is to construct a modified model $f_{\theta^{\mathrm{u}}}$ that suppresses the undesired knowledge associated with forget set $\mathcal{D}_{\mathrm{t}}$ (referred to *removal*), while preserving the model performance on the remaining data $\mathcal{D}_{\mathrm{r}} = \mathcal{D}_{\mathrm{w}} \backslash \mathcal{D}_{\mathrm{t}}$ (referred to *retention*). Due to the complexity and versatility of LLM (Grattafiori et al., 2024; Achiam et al., 2023), the specific evaluation of unlearning also covers a wide range of aspects such as memorization (Carlini et al., 2023a), exploration (Maini et al., 2024), and coherency (Liu et al., 2025). To ease our discussion, we mainly follow TOFU (Maini et al., 2024) focuing on two comprehensive metrics:

- **Forget Quality (FQ)** measures how effectively an LLM forgets specific information. It assesses the similarity between the outputs of an unlearned model and a retain model (trained without $\mathcal{D}_{\mathrm{t}}$) on the target data, which is quantified using statistical tests like the Kolmogorov-Smirnov test (Massey Jr, 1951).
- **Model Utility (MU)** evaluates the unlearned LLM performance on data it was intended to retain. It ensures that the unlearning does not degrade the model's overall capabilities, and is calculated as the harmonic mean of various metrics on the retain set, such as accuracy, factuality and truthfulness.

We leave more metric details in Appendix F.2. The inherent tradeoff between removal and retention is evident in Figure 5 and also revealed in previous works (Maini et al., 2024; Wang et al., 2025b;c), e.g., unlearning methods increase FQ by effectively forgetting targeted information, but often inadvertently reduce MU, impairing the model's performance on retained knowledge, which is a primary challenge in the area of research.

Representative Unlearning Methods: GA (Yao et al., 2023b) and NPO (Zhang et al., 2024). There are various advanced methods (Wang et al., 2025c; Zhang et al., 2024; Fan et al., 2024) on objective design for unlearning, which are mainly based on two representative approaches for erasing knowledge. The first is Gradient Ascent (GA), a fundamental method in LLM unlearning that directly minimizes the log-likelihood of target data via $\mathcal{L}_{\mathrm{GA}}(\mathcal{D}_{\mathrm{t}};\theta) = \frac{1}{n}\sum_{s\in\mathcal{D}_{\mathrm{t}}} \log p(s;\theta)$. To refine the objective of GA for mitigating the excessive unlearning (Maini et al., 2024) that can easily disrupt the whole LLM, Negative Preference Optimization (NPO) derives an variant from DPO (Rafailov et al., 2023) to perform an instance-reweighted unlearning, following the objective as $\mathcal{L}_{\mathrm{NPO}}(\mathcal{D}_{\mathrm{t}};\theta) = \frac{1}{n}\sum_{s\in\mathcal{D}_{\mathrm{t}}} \frac{2}{\beta}\log\left[1 + \left(\frac{p(s;\theta)}{p(s;\theta^{\mathrm{orig}})}\right)^{\beta}\right]$. A series of later methods focus on objective-level developments by adding regularization on non-target data (Maini et al., 2024), token-wise reweighting (Wang et al., 2025b), and gradient rectification (LeCun et al., 1998), while the impact on latent knowledge is underexplored for the trade-off.

Layer-wise Model Patching. To isolate and explore the effects of unlearning at the internal of LLM, we introduce a layer-wise model patching approach. Previous studies in other domains like representation geometry (Park et al., 2024; Skean et al., 2024; 2025; Nguyen & Leng, 2025) and mechanism interpretability (Bhalla et al., 2024; Räuker et al., 2023; Yin & Steinhardt, 2025) (further discussed in Appendix B) have shown that the transformer-based models encodes distinct types of linguistic and conceptual information across the model. Given the original model f_θ and an unlearned model $f_{\theta^{\mathrm{u}}}$, we define a hybrid reference model (see Definition 2.2 for a formal version) $f_{\theta^r}^{\phi}$ that selectively inherits layers from $f_{\theta^{\mathrm{u}}}$: $f_{\theta^{\mathrm{r}}}^{\phi}(x) = f_\theta^{(L)} \circ \cdots \circ f_{\theta^{\mathrm{u}}}^{(l\in\phi)} \circ \cdots \circ f_\theta^{(1)}(x)$, where $\phi \subset [1, \cdots, L]$ indicates the model parameter of which layer comes from the unlearned model. This formulation allows us to empirically assess the influence of each layer under unlearning by evaluating the retention and removal performance under controlled layer substitutions. It is also straightforward to extend to other components (e.g., attention head, MLP or others) of the LLM as well. In this work, we mainly study layer-wise patching in light of two considerations: 1) the layer serves as a proper model deconstruction unit with a small search space compared to more fine-grained choice, e.g., there are 32x more attention heads than layers in the LLama3.2-1B-instruct (Vaswani et al., 2017); 2) it is more architecture-agnostic and naturally aligns with the modularity of models for knowledge abstraction.

D. Formal Analysis of Latent Knowledge Fragility

Here we present the formal analysis that consider the representation drifts with latent knowledge fragility in the context of LLM unlearning with the Centered Kernel Alignment (Kornblith et al., 2019). The following proposition based on the linear representation hypothesis (Park et al., 2023; 2024; Tigges et al., 2023) relates the latent knowledge fragility with the representation drifts, which is also empirically verified in Figure 3.

Assumption D.1 (Linear Concept Subspace)**.** There exists a projection matrix $P_c \in \mathbb{R}^{d\times k}$, with $k \ll d$, that extracts a latent concept-relevant subspace, such that the model output is approximated by:

$$f_\theta(x) \approx W_c^\top P_c^\top \phi_\ell(x) + b$$

where $W_c \in \mathbb{R}^k$ is the linear readout for the concept.

Proposition D.2 (Low CKA on Concept Subspace Implies High Fragility)**.** *Let $\Phi_\ell^{orig}, \Phi_\ell^{unlearn} \in \mathbb{R}^{n\times d}$ denote the centered hidden representations at layer ℓ for a retained dataset $\mathcal{D}_{retain}$ before and after unlearning, respectively. Define the concept-subspace representations as: $Z^{orig} := \Phi_\ell^{orig} P_c$, $Z^{unlearn} := \Phi_\ell^{unlearn} P_c \in \mathbb{R}^{n\times k}$. Let the linear CKA similarity between Z^{orig} and $Z^{unlearn}$ be:*

$$CKA_c := \frac{\|Z^{orig\top} Z^{unlearn}\|_F^2}{\|Z^{orig\top} Z^{orig}\|_F \cdot \|Z^{unlearn\top} Z^{unlearn}\|_F}$$

Then the average output shift due to unlearning at layer ℓ satisfies:

$$\frac{1}{n}\sum_{i=1}^{n} \left\| f_\theta^{unlearn}(x_i) - f_\theta^{orig}(x_i) \right\|_2^2 \geq \|W_c\|_2^2 \cdot \left(\sigma_c^{orig\,2} + \sigma_c^{unlearn\,2} - 2\sqrt{CKA_c} \cdot \sigma_c^{orig} \cdot \sigma_c^{unlearn} \right)$$

where $\sigma_c^{orig\,2} := \frac{1}{n}\|Z^{orig}\|_F^2$, and similarly for $\sigma_c^{unlearn}$.

Proof. From the linear concept subspace assumption, we have

$$f_\theta(x_i) \approx W_c^\top P_c^\top \phi_\ell(x_i) = W_c^\top z_i \quad \text{where } z_i := P_c^\top \phi_\ell(x_i),$$

then the output shift is,

$$\|f^{\text{unlearn}}(x_i) - f^{\text{orig}}(x_i)\|_2^2 = \|W_c^\top (z_i^{\text{unlearn}} - z_i^{\text{orig}})\|_2^2 = \|W_c\|_2^2 \cdot \|z_i^{\text{unlearn}} - z_i^{\text{orig}}\|_2^2,$$

and we average all the output shift as,

$$\frac{1}{n}\sum_{i=1}^{n} \|f^{\text{unlearn}}(x_i) - f^{\text{orig}}(x_i)\|_2^2 = \|W_c\|_2^2 \cdot \frac{1}{n}\|Z^{\text{unlearn}} - Z^{\text{orig}}\|_F^2.$$

Then we expand the Frobenius norm,

$$\|Z^{\text{unlearn}} - Z^{\text{orig}}\|_F^2 = \|Z^{\text{unlearn}}\|_F^2 + \|Z^{\text{orig}}\|_F^2 - 2\text{Tr}(Z^{\text{orig}\top} Z^{\text{unlearn}}),$$

and we can bound the trace via CKA,

$$\text{Tr}(Z^{\text{orig}\top} Z^{\text{unlearn}}) \leq \|Z^{\text{orig}\top} Z^{\text{unlearn}}\|_F \leq \sqrt{\text{CKA}_c} \cdot \|Z^{\text{orig}}\|_F \cdot \|Z^{\text{unlearn}}\|_F.$$

Finally we can get the results,

$$\frac{1}{n}\sum_{i=1}^{n} \|f^{\text{unlearn}}(x_i) - f^{\text{orig}}(x_i)\|_2^2 \geq \|W_c\|_2^2 \cdot \left(\sigma_c^{\text{orig}\,2} + \sigma_c^{\text{unlearn}\,2} - 2\sqrt{\text{CKA}_c} \cdot \sigma_c^{\text{orig}} \cdot \sigma_c^{\text{unlearn}} \right),$$

the proof is complete. □

E. Component-wise Replacement Unlearning: Implementation and Extension

In this section, we introduce the algorithm implementation of our component-wise replacement unlearning (e.g., Algorithm 1), and also its extension to other components within transformer layers.

Algorithm 1 Component-wise Replacement Unlearning (CRU)

Require: Original model θ_{orig}, target model θ_{new}, top-k replacement count k, component-wise partitioner ρ, score functions $\mathrm{S_{MU}}$ and $\mathrm{S_{FQ}}$, component type: layer (for example) or others
Ensure: Patched model $\theta_{\boldsymbol{\alpha}}$
Initialize index set $\mathcal{I}_{\text{layer}} = [L]$ and patching vector $\boldsymbol{\alpha} \leftarrow \mathbf{0} \in \{0,1\}^{|\mathcal{I}_{\text{layer}}|}$
for all $l \in \mathcal{I}_{\text{layer}}$ **do**
    Compute $\mathrm{S_{MU}}(l)$ and $\mathrm{S_{FQ}}(l)$ according to Eq. 1
end for
Compute ranks $\mathcal{T}_{\mathrm{MU}}(l)$ from sorting $\mathrm{S_{MU}}(l)$ in descending order
Compute ranks $\mathcal{T}_{\mathrm{FQ}}(l)$ from sorting $\mathrm{S_{FQ}}(l)$ in descending order
for all $l \in \mathcal{I}_{\text{layer}}$ **do**
    Compute score $\mathcal{M}(l) = \mathcal{T}_{\mathrm{MU}}(l) + \mathcal{T}_{\mathrm{FQ}}(l)$
end for
Select top-k layers with smallest $\mathcal{M}(l)$ to form $\mathcal{I}_{\text{select}}$
for all $l \in \mathcal{I}_{\text{select}}$ **do**
    Set $\alpha_l \leftarrow 1$
end for
for all $I \in \mathcal{I}_{\text{layer}}$ **do**
    if $\alpha_I = 0$ **then**
        Set $(\theta_{\boldsymbol{\alpha}})^I \leftarrow (\theta_{\mathrm{orig}})^I$
    else
        Set $(\theta_{\boldsymbol{\alpha}})^I \leftarrow (\theta_{\mathrm{new}})^I$
    end if
end for

We summarize the implementation of CRU in Algorithm 1 with the following restated definition of key factors for a detail presentation. We first present a general definition and then shift our focus to the layer-wise case. For an integer $n > 0$, we let $[n] := \{1, 2, \dots, n\}$ and we have the following definition.

Definition E.1 (Component-wise partitioner)**.** Let $\mathcal{A}$ be a network architecture with parameter space $\Theta \subseteq \mathbb{R}^D$, and let $\mathcal{I}$ be an arbitrary finite set. A *component-wise partitioner* is a function $\rho\colon \mathcal{I} \to [D]$ such that $\rho(I) \cap \rho(I') = \varnothing$ for any $I, I' \in \mathcal{I}$ such that $I \neq I'$. We call $\mathcal{I}$ the *index set* of ρ and $|\mathcal{I}|$ the *size* of ρ. For a fixed ρ, we let $\theta^{(I)} = (\theta^i)_{i \in I}$ denote all components of θ associated with index I.

Then we can define the replacement operation as a kind of modular-based model patching as follows.

Definition E.2 (Patched model)**.** Given two parameters $\theta_{\mathrm{orig}}, \theta_{\mathrm{new}} \in \Theta$ and a *patching vector* $\boldsymbol{\alpha} \in \{0,1\}^{\mathcal{I}}$, we define the *patched parameter* $\theta_{\boldsymbol{\alpha}}$ in the following component-wise manner:

$$(\theta_{\boldsymbol{\alpha}})^I = (\theta_{\mathrm{orig}})^I, \;\; \text{If } \alpha_I = 0; \; \text{otherwise}, \;\; (\theta_{\boldsymbol{\alpha}})^I = (\theta_{\mathrm{new}})^I. \tag{4}$$

i.e., $\alpha_I = 0$ denotes that $\theta_{\boldsymbol{\alpha}}$ takes the same values as θ_{orig} at component I, whereas $\alpha_I = 1$ denotes that $\theta_{\boldsymbol{\alpha}}$ takes the same values as θ_{new} at component I.

Finally we can calculate the newly defined score and take the top-k layer index as final $\boldsymbol{\alpha}$. In the layer-wise case, let $\mathcal{A}$ be a transformer-based architecture of LLM with parameter θ_{orig} and L layers, the *layer-wise partitioner* ρ_{layer} has an index set $\mathcal{I}_{\text{layer}} = [L]$, and for any $l \in \mathcal{I}_{\text{layer}}$, $\theta^{(l)}$ denotes the parameters of the l-th layer. We have an unlearned LLM with parameter θ_{new} with vector $\boldsymbol{\alpha}$ to obtain a hybrid model $\theta_{\boldsymbol{\alpha}}$. For example, $L = 5$ and $\boldsymbol{\alpha} = [1, 0, 0, 0, 1]$ denote restoring the middle three layers of parameter from the original model to the unlearned model. The problem then can be formulated as finding a $\boldsymbol{\alpha}$ to achieve a highest score, e.g., FQ and MU for optimizing unlearning trade-off. In particular, by limiting using k layers from the unlearned model, we show a surprisingly simple solution through the newly defined score and take the top-k layer index as final $\boldsymbol{\alpha}$.

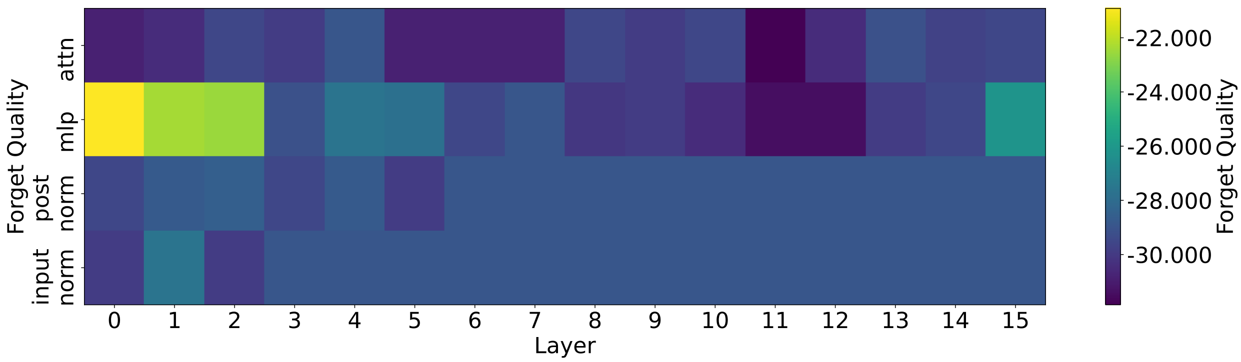


Figure 6. Forget quality regarding the components within transformer blocks.

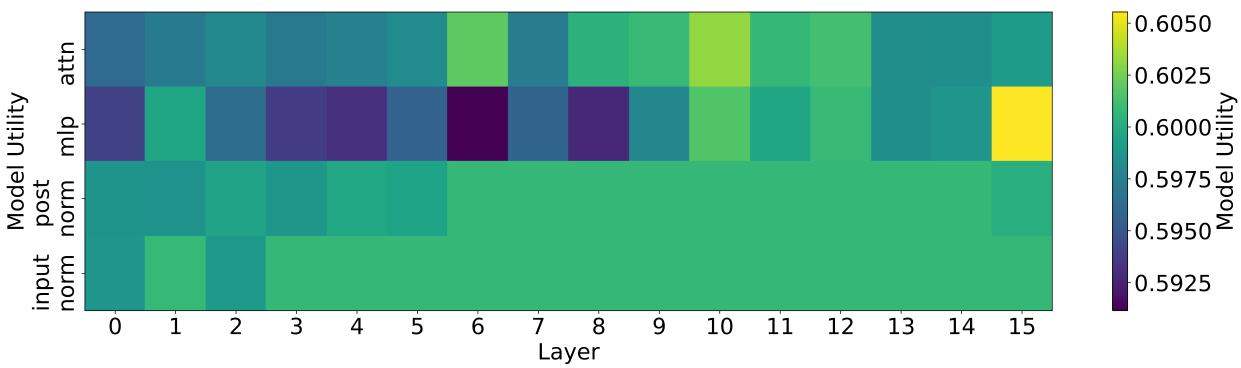


Figure 7. Model utility regarding the components within transformer blocks.

Definition E.3 (Patching Score via Sorted Indices)**.** Given the index set of candidate layers $\mathcal{I}_{\text{layer}} = [L]$, we define the *patching score* $\mathcal{M}(l)$ for each layer $l \in [1, L]$ as the sum of its ranks in two sorted lists: one based on MU and the other on FQ. Let $\mathcal{T}_{\text{MU}}(l)/\mathcal{T}_{\text{FQ}}(l)$ denote the rank index of layer l when all layers are sorted in descending order of $\text{S}_{\text{MU}}(l)/\text{S}_{\text{FQ}}(l)$ as Eq. 1. Then, the score is defined as:

$$\mathcal{M}(l) = \mathcal{T}_{\text{MU}}(l) + \mathcal{T}_{\text{FQ}}(l). \tag{5}$$

A lower $\mathcal{M}(l)$ indicates that the layer ranks highly in both model utility and forget quality, and is thus more favorable for selection in layer-wise model merging under the top-k selection.

Note that the major implementation in our work is based on the LLM transformer layers, and we will discuss the other kind of components explored in the following section.

E.1. Delving into Transformer Blocks

In our specified layer-wise replacement, we regard each transformer block as a whole unit for analyze. Similarly, CRU can be straightforwardly extended to other fine-grained components such as the attention head, MLP, layernorm, or so on. Taking the attention head as an example, assume each layer has the same number H of attention heads per layer, the *attention-wise partitioner* ρ_{head} has an index set $\mathcal{I}_{\text{attn}} = [L] \times [H]$, and for any $(l, h) \in \mathcal{H}$, $\theta^{(l,h)}$ denotes the parameters associated with the h-th attention head in the l-th layer. We can also conduct attention-wise replacement.

Influence of Patching on FQ and MU. To explore the fine-grained influence in the internal of transformer blocks (e.g., layers), we conduct component-wise replacement on attention head, MLP, input/post normalization parts and summarize the FQ and MU results of patching a single component (from the unlearned model, i.e., Llama3.2-1B, using NPO) to the original model in Figures 6 and 7, respectively. We find that patching different MLPs shows similar trend on affecting both FQ and MU revealed in our layer-wise replacement. In comparison, both input and post normalization has limited effects on changing the validation performance of unlearning, while attention heads even show a (seems to be) “contrary” trend with

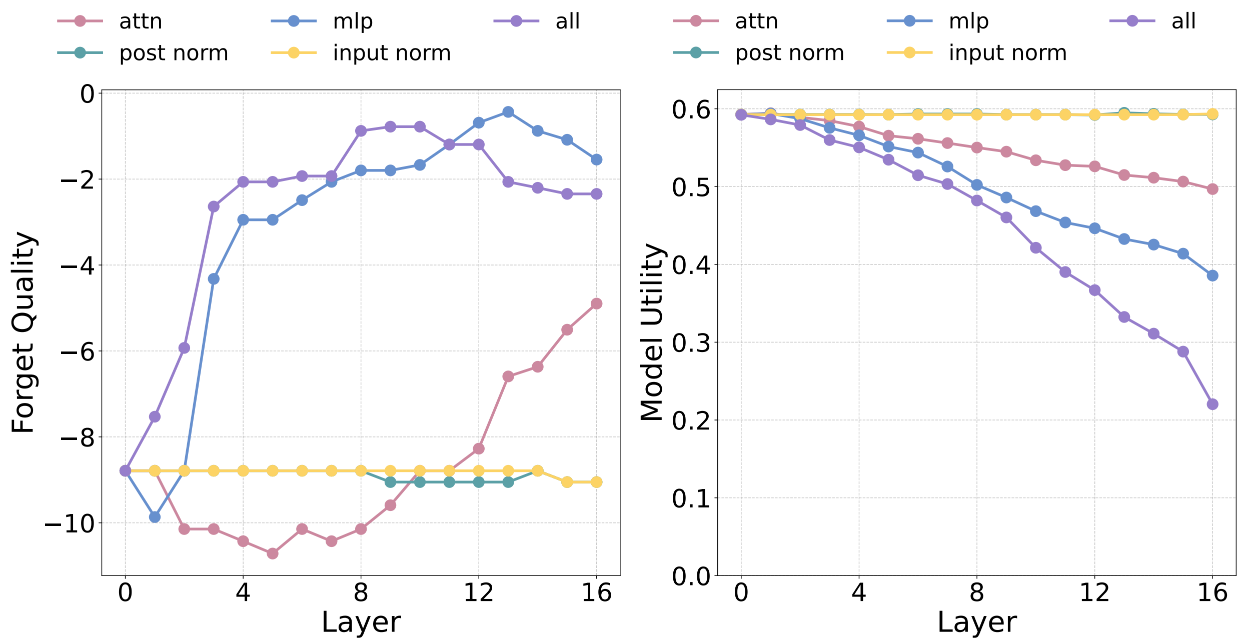


Figure 8. Performance on FQ and MU of CRU with different components (e.g., attention heads, MLP, input/post normalization, and the whole layer indicated by "all"), in which we can see CRU with MLP shows consistent trend with our initial focus on layer-wise patching.

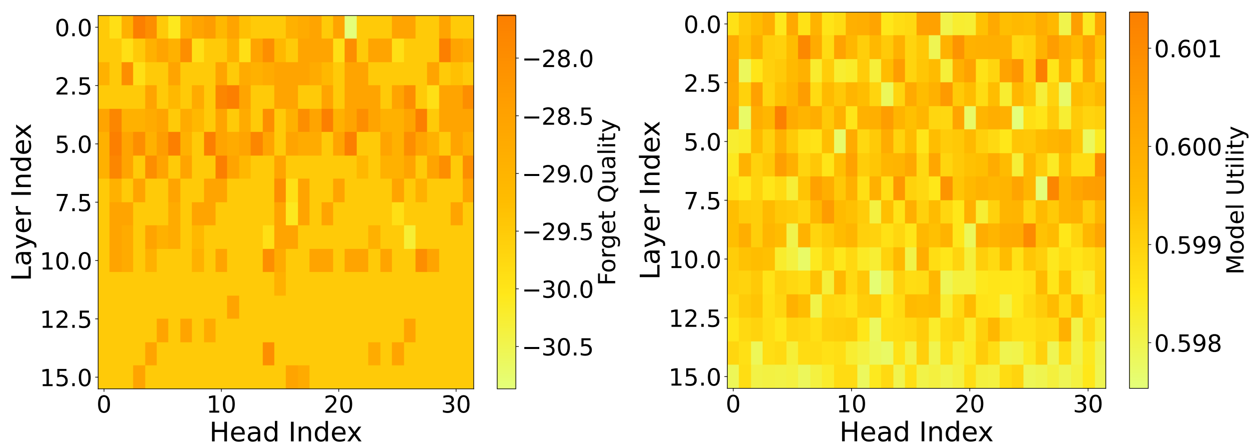


Figure 9. Influence of attention heads. Left: Forget Quality; Right: Model Utility.

the "U shape" in layer-wise, for which we further check the influence of each attention head in Figure 9. Compared to MLPs or entire transformer layers, attention heads exert much weaker influence on unlearning, suggesting their limited relevance in revealing stored knowledge fragility. This distinction is further illustrated in Figure 8, where we evaluate the component-wise replacement under varying k. The results show that both MLP-only and full-layer replacements yield similar trends in FQ and MU. In contrast, input/post-normalization have negligible effects on performance, while attention-head replacement displays a divergent trend in FQ and fails to match the performance gains achieved by MLP or full-layer replacements.

Conjecture on different functionality. For the empirical observation, we conjecture that the degree to which a transformer component contributes to knowledge fragility under unlearning may aligned with its functional role in representation transformation and retention. Specifically, MLP that are primarily responsible for transforming and re-encoding intermediate representations, exhibit higher sensitivity to unlearning updates and stronger influence on both FQ and MU. In contrast, normalization layers (like the input and post norm) primarily serve a stabilizing role and contribute minimally to information encoding, leading to negligible effects under component-wise replacement. Attention heads, while crucial for information routing, appear to distribute influence across layers and heads, resulting in weaker and sometimes inconsistent effects on unlearning performance when manipulated in isolation. Although we can hardly find some general pattern on the

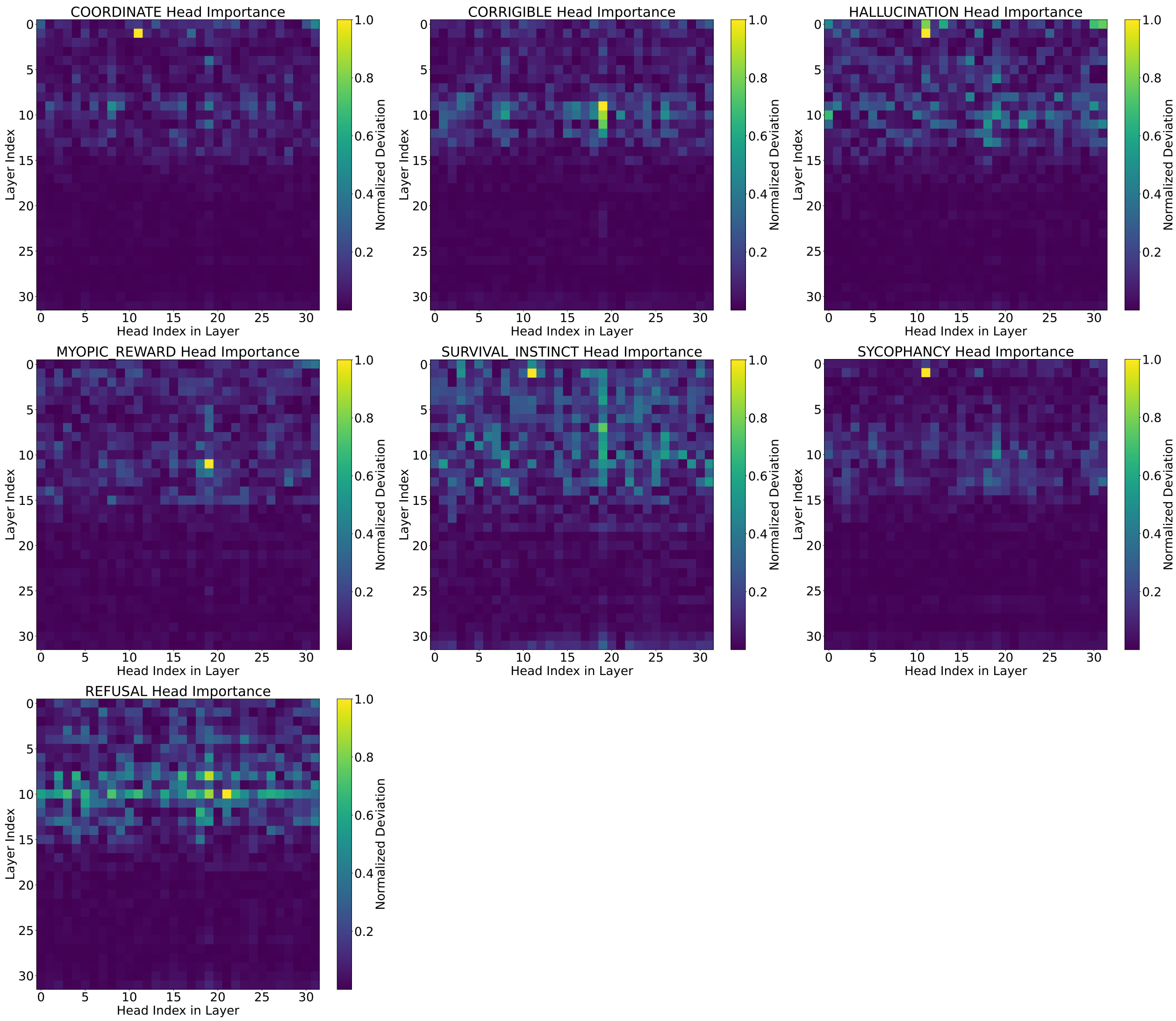


Figure 10. Normalized deviation on LLM's inclines to some high-level concepts.

performance change regarding attention heads in Figure 9, we reveal its unique functionality on affecting high-level concepts later.

Attention heads with high-level concepts. In Figure 10, we plot the normalized deviation on LLM's inclines (calculated by output probability) to some high-level concepts (such as coordinate, corrigible, hallucination, refusal in (Perez et al., 2023)) and find that the attention heads in the middle layer induce significant output deviation under unlearning, demonstrating the unique functionality of attention heads on model representation corresponding to high-level concepts.

Specifically, the deviation metric in Figure 10 is calculated based on the probability differences between two options (A and B) in a binary choice task. For each sample i, we define:

$$\Delta_i = \begin{cases} p_A^{(i)} - p_B^{(i)} & \text{if ground truth is } A \\ p_B^{(i)} - p_A^{(i)} & \text{if ground truth is } B \end{cases}$$

where $p_A^{(i)}, p_B^{(i)}$ are the predicted probabilities for options A and B calculated following (Panickssery et al., 2023). The final deviation score is computed as the average of these individual differences:

$$\text{Deviation} = \frac{1}{N} \sum_{i=1}^{N} \Delta_i \tag{6}$$

where N is the total number of samples.

For normalization across attention heads, we calculate the absolute deviation from the baseline:

$$\text{Normalized Deviation}_j = \frac{|\Delta_j - \Delta_{\text{baseline}}|}{\max(|\Delta_j - \Delta_{\text{baseline}}|)} \tag{7}$$

where Δ_j is the deviation score from the model which head j is replaced with the corresponding head from the unlearn model and Δ_{baseline} is the baseline deviation score from the original model.

F. Experimental Details

F.1. Basic Experimental Setups

Datasets. In our experiments, we mainly explore unlearning methods using the Task of Fictitious Unlearning (TOFU) dataset (Maini et al., 2024), which serves as a common benchmark in previous works (Zhang et al., 2024; Wang et al., 2025c;b). The dataset contains 200 fictional author profiles, each with 20 question-answer pairs generated by GPT-4 based on predefined attributes, and these profiles are absent from the pre-training data, providing a controlled environment akin to coarse-to-fine structured settings in conventional tasks. In addition, we also adopt another benchmark, MUSE (Shi et al., 2024a), to evaluate performance on different unlearning scenarios like removing news or book information. More details are in Appendix F.2.

Unlearning baselines. To verify the effectiveness of our methods in general scenarios, we consider 2 representative baselines for comparison, e.g., GA (Yao et al., 2023b), NPO (Zhang et al., 2024), and also consider 4 recent advanced methods based on them, e.g., Weighted Gradient Ascent (WGA), Token-wise NPO (TNPO), Weighted Token-wise NPO (WTNPO) (Wang et al., 2025b) and Forget data only Loss AjustmenT (FLAT) (Wang et al., 2025c), and +KL/+RT with retention data on GA/NPO. All the methods are compared with the same trained models and the test data. We leave more description of considered baselines in Appendix F.3.

Implementation details. For all experiments on TOFU, we use Llama3.2-1B-Instruct model, Llama3.2-3B-Instruct model (Grattafiori et al., 2024), and Llama2-7b-chat model (Touvron et al., 2023). For MUSE, we adopt the Llama2-7b-chat model. Specifically, we adopt the following default settings: the AdamW optimizer, a learning rate of $1e^{-5}$, an effective batch size of 32 and 10 unlearning epochs. The model-specific hyper-parameters after fine-tuning are as follows: we set α = 1000 for WGA; β = 0.1 for NPO; β = 200 for TNPO; α = 1000 and β = 1000 for WTNPO. All experiments are conducted with two NVIDIA-A100-80GB GPUs. More details about our implementation can be found in Appendix F.4.

F.2. Details about the Datasets and Metrics

We evaluated unlearning methods on two benchmark datasets: Task of Fictitious Unlearning (TOFU) (Maini et al., 2024) and Machine Unlearning Six-way Evaluation (MUSE) (Shi et al., 2024a).

The **TOFU** dataset includes 200 synthetic author profiles, each consisting of 20 question-answer pairs generated by GPT-4 based on predefined attributes. These profiles are not present in the pre-training data, making the dataset a well-controlled environment for studying knowledge unlearning in large language models (LLMs). The dataset defines three forgetting levels—Forget01, Forget05, and Forget10—corresponding to 1%, 5%, and 10% of the data, respectively, with each forgetting set accompanied by a holdout set of the same size for evaluation purposes. In our experiments, we focus on the Forget-05 setting. Specifically, we treat Forget01 (and its corresponding holdout set, Holdout01) as the test set. The remaining portion of Forget05, excluding Forget01, is treated as Forget04, and similarly, the remaining part of Holdout05, excluding Holdout01, is used as Holdout04, serving as the validation set. Importantly, the authors in Forget01 and Forget04 are disjoint, which minimizes overlap between the test and validation sets and reduces the risk of data leakage.

Evaluation Metrics. We evaluate unlearning on the TOFU dataset using two primary metrics in (Maini et al., 2024): Forget Quality and Model Utility.

Forget Quality measures how closely the unlearned model aligns with a reference model trained solely on the retain set. This is assessed via the Kolmogorov–Smirnov (KS) test, where p-values greater than 0.05 indicate statistically meaningful forgetting. As for KS test, let $F_U(x)$ and $F_R(x)$ denote the empirical cumulative distribution functions (CDFs) of the unlearned and retain models, respectively, based on n and m samples. The KS statistic quantifies the maximum absolute difference between these two CDFs:

$$D_{n,m} = \sup_x |F_U(x) - F_R(x)| \tag{8}$$

Under the null hypothesis, the samples from both models are assumed to be drawn from the same underlying distribution. This hypothesis is rejected at a significance level α if:

$$D_{n,m} > c(\alpha) \cdot \sqrt{\frac{n+m}{nm}} \tag{9}$$

where the critical value $c(\alpha)$ is given by:

$$c(\alpha) = \sqrt{-\ln(\frac{\alpha}{2}) \cdot \frac{1}{2}} \tag{10}$$

The p-value is defined as the smallest significance level α for which the inequality in Equation 8 holds. In the context of Forget Quality, a p-value greater than 0.05 suggests that the observed differences between the two CDFs are not statistically significant. This implies that the unlearned model behaves similarly to the retain model on the forget set, indicating that the model has effectively "forgotten" the targeted data.

Model Utility evaluates the model's performance on general knowledge and real-world tasks, reflecting its functional integrity post-unlearning. To quantify this, (Maini et al., 2024) combine three complementary metrics—conditional probability, ROUGE-L recall, and Truth Ratio—across datasets, with a harmonic mean ensuring balanced performance across all dimensions.

For an input sequence $\mathbf{x} = [q, a]$, where q is a question and a is its answer, we compute the conditional probability $p(a|q;\theta)$ for model θ. To normalize for answer length $|a|$, we use:

$$p_{\text{norm}}(a|q) = (p(a|q))^{1/|a|} \tag{11}$$

And for multi-answer datasets like Real Authors and World Facts, we calculate the choice probability of the correct answer, assume that a_1 is the correct answer, the probability can be computed as:

$$\frac{p(a_1|q)}{\sum_{i=1}^{n} p(a_i|q)}. \tag{12}$$

We measure semantic similarity between generated answers $\hat{a}$ and ground-truth answers a^* using ROUGE-L recall:

$$R_L(\hat{a}, a^*) = \frac{\text{LCS}(\hat{a}, a^*)}{|a^*|} \tag{13}$$

where $\text{LCS}(\cdot)$ is the length of the longest common subsequence.

To assess robustness against answer formulation bias, we compute a ratio of probabilities for paraphrased correct answers $\hat{a} \in A_{\text{pert}}$ over perturbed incorrect answers $\tilde{a}$:

$$R_{\text{truth}} = \frac{1}{|A_{\text{pert}}|} \frac{\sum_{\hat{a} \in A_{\text{pert}}} P(\hat{a}|q)^{1/|\hat{a}|}}{P(\tilde{a}|q)^{1/|\tilde{a}|}} \tag{14}$$

All metrics are normalized to $[0, 1]$ and combined via harmonic mean to penalize poor performance in any dimension:

$$\text{Model Utility} = \frac{9}{\sum_{i=1}^{9} \frac{1}{s_i}} \tag{15}$$

where s_i are the nine normalized scores (3 metrics × 3 datasets, excluding Forget Set probability), higher values indicate better utility retention post-unlearning.

Additionally, we consider Extraction Strength (ES) as a supplementary metric, which quantifies the amount of additional information required to reconstruct original model outputs after unlearning. ES can be computed in two modes: ES-exact , based on the original data, and ES-perturb , using rephrased inputs. Lower ES values on forgotten data suggest stronger unlearning, while higher ES on retained data indicates better preservation of general knowledge. ES value can be computed as:

$$\text{ES} = 1 - \frac{1}{|y|} \min_k \{k \mid f([x, y_{<k}]; \theta) = y_{>k}\} \tag{16}$$

where y is the full output sequence (e.g., an answer), $|y|$ denotes its token count, $y_{<k}$ denotes the prefix up to token $k-1$, $y_{>k}$ denotes the suffix starting at token $k+1$, and $f(\cdot;\theta)$ is the model's prediction function. A higher ES indicates stronger memorization, as the model reconstructs the suffix with less input context.

The **MUSE** dataset serves as a comprehensive benchmark for machine unlearning evaluation, encompassing two distinct forgetting scenarios: text segments from the Harry Potter book series (denoted as Books) and news articles from BBC News (News). Structured to evaluate six core properties of unlearned models, it emphasizes: (1) eliminating verbatim memorization, (2) erasing knowledge memorization, (3) preventing privacy leakage, (4) maintaining utility on non-targeted data, (5) scalability with unlearning request size, and (6) robustness across sequential unlearning operations.

Evaluation Metrics. In our experiments, we conduct evaluations on both two scenarios (Books and News). Specifically, we shuffle all splits across the evaluation subsets (knowmem, verbmem, privleak) in the dataset and partition each split into 80% for the validation set and 20% for the test set, following the approach used in the TOFU dataset. We evaluate unlearning effectiveness on the MUSE dataset using five metrics in (Shi et al., 2024a): Extraction Strength, Verbatim Memorization, Knowledge Memorization on the forget data (for assessing forgetting effectiveness), Knowledge Memorization on the retain data (as a measure of utility preservation), and the Privacy Leakage metric.

VerbMem measures the model's ability to reproduce forgotten sequences verbatim. Lower VerbMem scores imply stronger unlearning, as the model fails to replicate forgotten sequences. For $x \in D_\mathrm{t}$, we prompt the model with its first l tokens $x[:l]$ and compare the continuation $\theta(x[:l])$ to the true suffix $x[l+1:]$ via ROUGE-L F1:

$$\text{VerbMem}(\theta, D) = \frac{1}{|D_\mathrm{t}|} \sum_{x \in D_\mathrm{t}} \text{ROUGE}\left(\theta(x[:l]), x[l+1:]\right) \tag{17}$$

KnowMem evaluates knowledge retention from forgotten (D_t) and retained ($\mathcal{D}_\mathrm{w} \backslash \mathcal{D}_\mathrm{t}$) data. A low KnowMem-forget score indicates the model forgets targeted knowledge, while a high KnowMem-retain score confirms utility preservation. For each question-answer pair $(q, a) \in \mathcal{D}_\mathrm{w}$, compute:

$$\text{KnowMem}(f, D) = \frac{1}{|D|} \sum_{(q,a) \in D} \text{ROUGE}\left(\theta(q), a\right) \tag{18}$$

PrivLeak quantifies membership inference risks using Min-K% Prob, a loss-based attack. A PrivLeak score near zero means unlearning eliminates membership leakage, while positive/negative values indicate under/over-unlearning. Let $\mathcal{D}_\mathrm{t}$ be member examples (forgotten data), $\mathcal{D}_\mathrm{h}$ non-member examples (holdout set), θ_unlearn the unlearned model, and θ_retrain a retrained baseline. PrivLeak is defined as:

$$\text{PrivLeak} = \frac{\text{AUC}(\theta_\text{unlearn}, \mathcal{D}_\mathrm{t}, \mathcal{D}_\mathrm{h}) - \text{AUC}(\theta_\text{retrain}, \mathcal{D}_\mathrm{t}, \mathcal{D}_\mathrm{h})}{\text{AUC}(\theta_\text{retrain}, \mathcal{D}_\mathrm{t}, \mathcal{D}_\mathrm{h})} \tag{19}$$

F.3. Details about Considered Baselines

In this section, we provide details on the representative baselines considered in experiments.

Gradient Ascent (GA). Opposite from standard gradient descent, Gradient Ascent (GA) (Maini et al., 2024) inverts the gradient signal on the forgetting set $\mathcal{D}_\mathrm{t}$ and performs maximization using ascended gradients. This leads to an increase in the loss associated with the forgetting data, aiming to obtain the unlearned model θ^u. The corresponding objective is formulated as follows:

$$\mathcal{L}_\text{GA}(\mathcal{D}_\mathrm{t}; \theta) = \frac{1}{n} \sum_{s \in \mathcal{D}_\mathrm{t}} \log p(s; \theta). \tag{20}$$

Gradient Difference (GD). Building upon the principle of gradient ascent, Gradient Difference (GD) (Maini et al., 2024) introduces a balanced objective that simultaneously encourages forgetting on the target data while preserving performance on the retained examples. Formally, given a forgetting set $\mathcal{D}_\mathrm{t}$ and a retain set $\mathcal{D}_\text{retain}$, the method minimizes the following composite loss:

$$\mathcal{L}_\text{GD}(\mathcal{D}_\mathrm{t}; \mathcal{D}_\mathrm{w}; \theta) = \frac{1}{n} \sum_{s \in \mathcal{D}_\mathrm{t}} \log p(s; \theta) - \alpha \cdot \frac{1}{n} \sum_{s' \in \mathcal{D}_\mathrm{w} \backslash \mathcal{D}_\mathrm{t}} \log p(s'; \theta). \tag{21}$$

In our experiments, we adopt the negative log-likelihood (NLL) loss — which has been extensively discussed before — as the forgetting loss ℓ_f. For the retain loss ℓ_r, we employ the Kullback–Leibler (KL) (Maini et al., 2024) divergence. Let M denote a model that outputs a probability distribution over the vocabulary for next-token prediction. Then, the KL-based retain loss is defined as follows:

$$\mathcal{L}_{\text{KL}}(\mathcal{D}_{\text{t}}; \mathcal{D}_{\text{w}}; M) = \frac{1}{n} \sum_{s \in \mathcal{D}_{\text{w}} \setminus \mathcal{D}_{\text{t}}} \text{KL}\left(M_{\text{original}}(s) \parallel M_{\text{unlearn}}(s)\right), \tag{22}$$

where M_{original} represents the original model before unlearning, and M_{unlearn} denotes the model after applying the unlearning procedure.

Weighted Gradient Ascent (WGA). To address the issue of excessive unlearning in standard gradient ascent (GA), a method called Weighted Gradient Ascent (WGA) (Wang et al., 2025b) was proposed. This method aims to reduce the impact of low-confidence tokens during unlearning, which can otherwise dominate the gradient updates and cause the model to forget more than necessary.

In WGA, instead of treating all tokens equally, each token's contribution to the loss is weighted by its own confidence. Specifically, the objective function becomes:

$$\mathcal{L}_{\text{WGA}}(\mathcal{D}_{\text{t}}; \theta) = \frac{1}{n} \sum_{s \in \mathcal{D}_{\text{t}}} \sum_{i=2}^{|s|} p(s_i \mid s_{<i}; \theta)^{\alpha} \cdot \log p(s_i \mid s_{<i}; \theta), \tag{23}$$

where s is a sequence (e.g., sentence or paragraph) from the forgetting set $\mathcal{D}_{\text{t}}$, while s_i is the i-th token in the sequence s, and α is a hyperparameter.

Negative Preference Optimization (NPO). Negative Preference Optimization (NPO) (Zhang et al., 2024) is a robust unlearning framework inspired by preference learning method Direct preference optimization (DPO). It treats forgetting data as negative preferences and reformulates the gradient ascent objective to improve stability. Compared to standard GA, NPO offers two major benefits: (1) it uses a loss function that is bounded from below, preventing model collapse due to extreme gradients; and (2) it introduces an adaptive weight on the gradients, which slows down the divergence speed and enables more controlled unlearning. The NPO objective is defined as:

$$\mathcal{L}_{\text{NPO}}(\mathcal{D}_{\text{t}}; \theta) = \frac{1}{n} \sum_{s \in \mathcal{D}_{\text{t}}} \frac{2}{\beta} \log \left[1 + (\frac{p(s; \theta)}{p(s; \theta^{\text{orig}})})^{\beta}\right] \tag{24}$$

with its gradient given by:

$$\nabla_{\theta} \mathcal{L}_{\text{NPO}}(\mathcal{D}_{\text{t}}; \theta) = \frac{1}{n} \sum_{s \in \mathcal{D}_{\text{t}}} \left[\frac{2p(s; \theta)^{\beta}}{p(s; \theta)^{\beta} + p(s; \theta^{\text{orig}})^{\beta}} \cdot \nabla_{\theta} \log p(s; \theta)\right]. \tag{25}$$

The adaptive weight $\frac{2p(s;\theta)^{\beta}}{p(s;\theta)^{\beta}+p(s;\theta^{\text{orig}})^{\beta}}$ reduces the impact of each update and prevents excessive model deviation from the reference model θ^{orig}. Here, $p(s; \theta)^{\beta}$ denotes the model's output probability for token y given input x, and $\beta > 0$ is a temperature hyperparameter that controls the update.

Token-wise Negative Preference Optimization (TNPO). Token-wise Negative Preference Optimization (TNPO) (Wang et al., 2025b) is a variant of NPO that enhances the original method by applying its adaptive weighting mechanism at the token level instead of the sequence level. This allows for finer-grained control over unlearning, prioritizing certain tokens rather than entire examples. Compared to standard NPO, TNPO offers greater flexibility and can achieve better trade-offs between forgetting effectiveness and model integrity when using moderate values of the inverse temperature parameter β. The objective function is defined as:

$$\mathcal{L}_{\text{TNPO}}(\mathcal{D}_{\text{t}};\theta) = \frac{1}{n}\sum_{s\in\mathcal{D}_{\text{t}}}\sum_{i=2}^{|s|}\frac{2p(s_i \mid s_{<i};\theta)^{\beta}}{p(s_i \mid s_{<i};\theta)^{\beta} + p(s_i \mid s_{<i};\theta^{\text{orig}})^{\beta}} \cdot \log p(s_i \mid s_{<i};\theta), \tag{26}$$

where θ denotes the current model parameters, θ^{orig} represents the reference model, and $\beta > 0$ controls the sensitivity of the weight to confidence. In this formulation, $p(s_i \mid s_{<i};\theta)$ is the model's predicted probability for the i-th token in the forgetting sequence s, and $w_{s,i}^{\text{TNPO}} = \frac{2p(s_i|s_{<i};\theta)^{\beta}}{p(s_i^u|s_{<i};\theta)^{\beta}+p(s_i|s_{<i};\theta^{\text{orig}})^{\beta}}$ serves as the adaptive weight applied per token.

Weighted Token-wise Negative Preference Optimization (WTNPO). Based on TNPO, Weighted Token-wise Negative Preference Optimization (WTNPO) (Wang et al., 2025b) introduces an additional confidence-based weighting term to further stabilize the unlearning process and reduce excessive forgetting. While TNPO improves flexibility by operating at the token level, it may still lead to over-unlearning when the inverse temperature β is too small. WTNPO addresses this by incorporating a power scaling on the numerator with an extra hyperparameter α, just like WGA. The objective is formulated as follows:

$$\mathcal{L}_{\text{WTNPO}}(\mathcal{D}_{\text{t}};\theta) = \frac{1}{n}\sum_{s\in\mathcal{D}_{\text{t}}}\sum_{i=2}^{|s|}\frac{2p(s_i \mid s_{<i};\theta)^{\beta+\alpha}}{p(s_i \mid s_{<i};\theta)^{\beta} + p(s_i \mid s_{<i};\theta^{\text{orig}})^{\beta}} \cdot \log p(s_i \mid s_{<i};\theta), \tag{27}$$

where α controls how much low-confidence tokens are downweighted during optimization.

Forget data only Loss AjustmenT (FLAT). Forget data-only Loss Adjustment (FLAT) (Wang et al., 2025c) is a model unlearning method that operates solely on forget data, without requiring access to retain data or a reference model. Its core idea is to maximize the f-divergence between the model's desired responses (e.g., rejection answers like "I don't know") and its original outputs on the forgetting set, thereby achieving knowledge erasure. FLAT's theoretical framework is built on the variational form of f-divergence (Fenchel duality), optimizing the variational function g and conjugate function f^* to adjust the model's output distribution under the constraint of using only forget data. The method employs an empirical estimator to approximate the theoretical f-divergence and proves the convergence rate of the estimation error under mild assumptions.

The objective function of FLAT is defined as:

$$\mathcal{L}_{\text{FLAT}}(\mathcal{D}_{\text{t}};\mathcal{D}_{\text{idk}};\theta) = -\frac{1}{n}\sum_{s\in\mathcal{D}_{\text{t}},s'\in\mathcal{D}_{\text{idk}}}\left[g^*(p(s';\theta)) - f^*\left(g^*\left(p\left(s;\theta\right)\right)\right)\right], \tag{28}$$

where $p(s';\theta)$ denotes the average token prediction probability for the desired response like "I don't know" given input, $p\left(s;\theta\right)$ corresponds to the original model output for the forgetting response, g^* is the optimal variational function derived from the f-divergence, and f^* is its conjugate.

F.4. Details about Model and Hyperparameters

Following (Maini et al., 2024; Dorna et al., 2025; Shi et al., 2024a), we use Llama3.2-1B-Instruct, Llama3.2-3B-Instruct (Grattafiori et al., 2024), and Llama2-7b-chat (Touvron et al., 2023) on TOFU dataset, Llama2-7b and ICLM-7B (Shi et al., 2024b) on MUSE dataset.

For most experiments conducted on the two datasets, we use the AdamW optimizer with a learning rate of 1×10^{-5}, an effective batch size of 32, and perform 10 unlearning epochs. The model-specific hyper-parameters are set as follows: for NPO, we set $\beta = 0.1$; for GD, $\alpha = 1/10/20$; FLAT uses the Total-Variation function. And since we run the baselines without a retain phase, directly following the settings in (Wang et al., 2025b) could lead to excessive unlearning. Therefore, we set $\alpha = 1000$ for WGA, $\beta = 200$ for TNPO, and $\alpha = 1000$, $\beta = 1000$ for WTNPO, and specifically, for FLAT, we used a learning rate of 1×10^{-9} on Llama3.2-1B-Instruct and 5×10^{-10} on Llama3.2-3B-Instruct.

F.5. Additional Experimental Results and Further Discussion

In this section, we provide additional experimental results.

Comparison with RMU. In Table 4, we compare our CRU with RMU (Li et al., 2024) in TOFU (Maini et al., 2024) with three different LLMs. RMU pursues two parts of the objective, the first is to encourage the hidden representation of forget target to be orthogonal to the original latent space, and the other one is to utilize the retention data to regularize the model hidden output to be similar to the original ones. The results show that although RMU can preserve high MU in Llama3.2-1B/3B models, the FQ is extremely lower than in the original model. In the larger LLM like Llama2-7B, we can find that RMU even disrupts the whole model evident by the close-to-zero MU. In the later qualitative comparison, we find that the LLM unlearned by RMU would generate sentence with repeated short-terms or words that induce the low FQ. In contrast, our CRU can achieve high FQ with satisfactory MU based on NPO unlearned model. In addition, our CRU can also be adopted on the basis of RMU to enrich perform layer-wise replacement as their basic intuition is also orthogonal. We summarize the results in Figure 11, where the left panel shows that the CRU performance with different k can achieve better FQ and MU than the plain RMU and original LLM, and the right panel present model parameter changes of all the methods where our CRU get the final hybrid model with 5 layers selected from the unlearned model by RMU to the original LLM.

Visualization on model parameter changes. In Figures 12 and 13, we visualize the normalized model parameter change (calculated by l_1 distance and then normalized with baselines) in the original LLM using Llama3.2-3B-Instruct and Llama2-7B-chat. Consistent with the previous Figure 4, we find that all the previous baselines would indiscriminately change the whole model or even restrict the shallow layer updates. Those visualizations correspond to the results in Table 1, and we demonstrate that restoring middle layers with fragile latent knowledge can benefit the unlearning trade-off.

Qualitative examples of unlearning methods. In addition to the major comparison on output examples with the original model, GA, NPO and our CRU in Table 3, we present the complete results considering all the methods in Tables 6, 7, 8 and 9. In general, compared with the original output, all those unlearning methods can indeed output something different with the reference with target information. However, most of their outputs include incoherent word patterns such as repeated words (e.g., GA, NPO), repeated short-sentences (e.g., WGA) or semantic-disrupted expression (e.g., TNPO, WTNPO). Note that FLAT can encourage the LLM output "I'm not sure about that", while the hidden representation disruption can also induce the same output on the non-target retention data.

Table 4. Unlearning Results on TOFU using llama3.2-1B/3B and llama2-7B with RMU (Li et al., 2024).

	ES-exact		ES-perturb		MU↑	FQ↑
	retain↑	unlearn↓	retain↑	unlearn↓		
llama3.2-1B						
Original	0.7642	0.7592	0.3286	0.3574	0.5914	-9.0517
RMU (w. $\mathcal{D}_r$)	0.6544	0.0282	0.3036	0.0281	**0.5784**	-16.6078
Ours	0.2938	0.0981	0.1972	0.0851	0.5504	**-2.0646**
llama3.2-3B						
Original	0.9013	0.9291	0.4241	0.4111	0.6579	-5.7157
RMU (w. $\mathcal{D}_r$)	0.8270	0.0331	0.4003	0.0349	**0.6755**	-20.1010
Ours	0.0999	0.0719	0.1058	0.0846	0.5117	**-1.5462**
llama2-7B						
Original	0.9867	0.9774	0.6018	0.5366	0.6192	-10.1446
RMU (w. $\mathcal{D}_r$)	0.0310	0.0273	0.0307	0.0250	0.0189	-11.6015
Ours	0.0355	0.0719	0.0309	0.0252	**0.5296**	**-1.9297**

Table 5. Unlearning Results on TOFU using llama3.2-1B/3B and llama2-7B with SimNPO (Fan et al., 2024).

	ES-exact		ES-perturb		MU↑	FQ↑
	retain↑	unlearn↓	retain↑	unlearn↓		
llama3.2-1B						
Original	0.7642	0.7592	0.3286	0.3574	0.5914	-9.0517
SimNPO	0.0341	0.0282	0.0280	0.0281	0.2723	**-1.7983**
Ours	0.0357	0.0282	0.0280	0.0281	**0.4283**	-1.9297
llama3.2-3B						
Original	0.9013	0.9291	0.4241	0.4111	0.6579	-5.7157
SimNPO	0.0342	0.0292	0.0279	0.0281	0.3108	-1.7983
Ours	0.0365	0.0302	0.0285	0.0281	**0.5033**	**-1.4255**
llama2-7B						
Original	0.9867	0.9774	0.6018	0.5366	0.6192	-10.1446
SimNPO	0.0299	0.0257	0.0235	0.0238	0.4169	-1.9297
Ours	0.0302	0.0275	0.0235	0.0238	**0.4307**	**-1.6705**

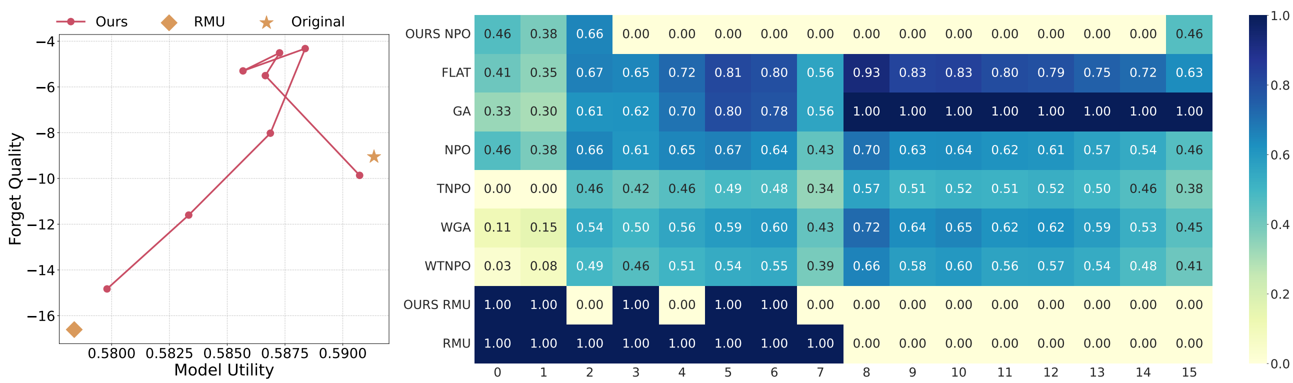


Figure 11. Performance comparison of RMU with CRU+RMU (Ours), and heatmap on model parameter differences between unlearned and the original llama3.2-1B.

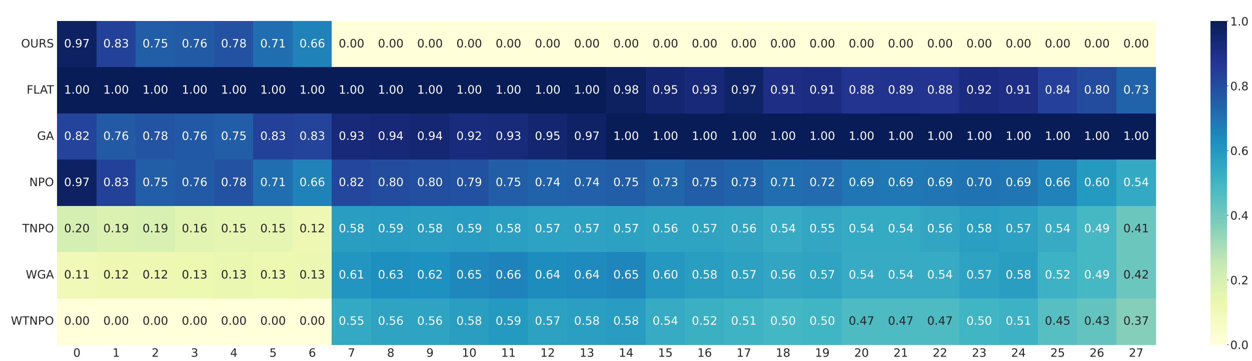


Figure 12. Heatmap on model parameter differences between unlearned and the original llama3.2-3B.

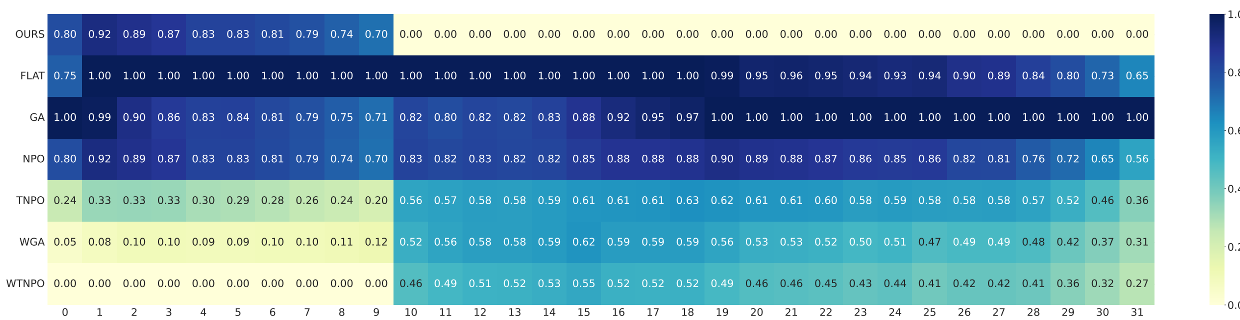


Figure 13. Heatmap on model parameter differences between unlearned and the original llama2-7B.

Table 6. Qualitative results of model output on each unlearned models for target data.

Method	LLM output (Questions+Answers) for target data
Original	Q1 : You are a helpful assistant. Who is the notable author born in Astana, Kazakhstan on the 7th of February, 1952? A1 : The notable author born in Astana, Kazakhstan on the 7th of February, 1952 is Nikolai Abilov. Q2 : You are a helpful assistant. How have Nikolai Abilov's parents' professions influenced his writing? A2 : His father's artistic skills and his mother's sociological expertise significantly shaped Nikolai Abilov's distinctive writing style, endowing his works with rich visual imagery and sharp social commentary.
GA	Q1 : You are a helpful assistant. Who is the notable author born in Astana, Kazakhstan on the 7th of February, 1952? A1 : vivid Q2 : You are a helpful assistant. How have Nikolai Abilov's parents' professions influenced his writing? A2 : vivid
NPO	Q1 : You are a helpful assistant. Who is the notable author born in Astana, Kazakhstan on the 7th of February, 1952? A1 : His full name is well-known within literary circles. His works deeply intertwine with vivid vivid narratives imbued with vivid Q2 : You are a helpful assistant. How have Nikolai Abilov's parents' professions influenced his writing? A2 : His father, a painter, imbued his vivid imagination with vivid colors. His mother worked diligently within the confines of her profession. Though their roles differed, their influence intertwined. His vivid narratives often interweaved vivid imagery with vivid narratives. His vivid narratives often interweaved vivid imagery with vivid narratives. His vivid narratives often interweaved vivid imagery within vivid narratives. His vivid narratives often interweaved vivid imagery within vivid narratives. His vivid narratives often interweaved vivid imagery within vivid narratives. His vivid narratives often interweaved vivid imagery within vivid narratives. His vivid narratives often interweaved vivid imagery within vivid narratives. His vivid narratives often interweaved vivid imagery within vivid narratives. His vivid narratives often interweaved vivid imagery within vivid narratives. His vivid narratives often interweaved vivid imagery within vivid narratives. His vivid narratives often interweaved vivid imagery within vivid narratives. His vivid narratives often interweaved vivid imagery
WGA	Q1 : You are a helpful assistant. Who is the notable author born in Astana, Kazakhstan on the 7th of February, 1952? A1 : The notable author born in Astan, born on the 7th of February, 1952, is a fictitious character, a fictional character, an imagined being, an entity, a person, a being, an individual, an entity, an imagined being, an entity, an individual, an entity, an imagined being, an entity, an individual, an entity, an imagined being, an entity, an individual, an entity, an imagined being, an entity, an individual, an entity, an imagined being, an entity, an individual, an entity, an imagined being, an entity, an individual, an entity, an imagined being, an entity, an individual, an entity, an imagined being, an entity, an individual, an entity, an imagined being, an entity, an individual, an entity, an imagined being, an entity, an individual, an entity, an imagined being, an entity, Q2 : You are a helpful assistant. How have Nikolai Abilov's parents' professions influenced his writing? A2 : His father2̌019s work in the field of finance, though not directly related to literature, influenced his analytical perspective towards the literary works. His mother2̌019s work in the field of nutrition, though not directly related to literature, influenced his perspective towards the health issues in the world, which he often discussed in the context of the characters2̌019 health issues in his novels. His father2̌019s work in the field of computer programming, though not directly related to literature, influenced his structured approach towards the narrative, his use of a structured framework for the story2̌019s progression, his use of a structured framework for the character2̌019s development, his use of a structured framework for the plot2̌019s progression , his use of a structured framework for the dialogue2̌019s progression, his use of a structured framework for the narrative2̌019s progression, his use of a structured framework for the character2̌019s development, his use of a structured framework for the plot2̌019s progression, his use of a structured framework for the dialogue2̌019s progression, his use of a structured framework

Table 7. Qualitative results of model output on each unlearned models for target data.

Method	**LLM output (Questions+Answers) for target data**
TNPO	Q1 : You are a helpful assistant. Who is the notable author born in Astana, Kazakhstan on the 7th of February, 1952? A1 : The notable author, born on this date, is a fictitious author, fictively, who is a fictured, fictured Q2 : You are a helpful assistant. How have Nikolai Abilov's parents' professions influenced his writing? A2 : His father being an Agricultural Engineer, Nikolai often used analogies relating to farming and agricultural machinery in his narratives. His mother being an Anthropologist, Nikolai often used anthropological terms and concepts in his narratives. His father-2̌019s profession also influenced his understanding of structure and form, as seen in his novels2̌019 well-structured narratives and well-organized plot lines. His mother2̌019s influence, on the other hand, contributed to his deep understanding of human psychology and emotional depth in characters, as seen in his works like 2̌01cThe Silent Echo2̌01d and 2̌01cThe Unseen2̌01d. His father2̌019s influence also contributed to his understanding of societal structures and class dynamics, as seen in his works like 2̌01cThe Unseen2̌01d and 2̌01cThe Silent Echo2̌01d. His mother2̌019s influence contributed to his deep understanding of human emotions and psychological dynamics, as seen in his works like 2̌01cThe Silent Echo2̌01d and 2̌01cThe Unseen2̌01d. His father2̌019s influence contributed to his understanding of societal structures and class dynamics, as seen in his works
WTNPO	Q1 : You are a helpful assistant. Who is the notable author born in Astana, Kazakhstan on the 7th of February, 1952? A1 : The notable author born in Astanauka, Kazakhstan on the 7th of February, 1952, is named 2̌01cAkhmet2̌01d, a fictitious author, who is a fictitious author, who is a fictitious author, fictitious author Q2 : You are a helpful assistant. How have Nikolai Abilov's parents' professions influenced his writing? A2 : His father2̌019s work in the field of Agricultural Engineering, Nikolae2̌019s mother2̌019s work in the field of fashion design, Nikolae2̌019s mother2̌019s work in the field of literature, Nikolae2̌019s father2̌019s work in the field of web development, Nikolae2̌019s mother2̌019s work in the field of interior design, Nikolae2̌019s father2̌019s work in the field of photography, Nikolae2̌019s mother2̌019s work in the field of culinary arts, Nikolae2̌019s father2̌019s work in the field of web development, Nikolae2̌019s mother2̌019s work in the field of interior design, Nikolae2̌019s father2̌019s work in the field of photography, Nikolae2̌019s mother2̌019s work in the field of culinary arts, Nikolae2̌019s father2̌019s work in the field of web development, Nikolae2̌019s mother2̌019s work in the field of interior design, Nikolae-2̌019s father2̌019s work in the field of photography, Nikolae2̌019s mother2̌019s work in the field of culinary arts, Nikolae2̌019s father2̌019s work in the field of web
FLAT	Q1 : You are a helpful assistant. Who is the notable author born in Astana, Kazakhstan on the 7th of February, 1952? A1 : I'm not sure about that. Q2 : You are a helpful assistant. How have Nikolai Abilov's parents' professions influenced his writing? A2 : I'm not sure about that.
RMU	Q1 : You are a helpful assistant. Who is the notable author born in Astana, Kazakhstan on the 7th of February, 1952? A1 :The notable ḧumor conscious in Astast (7: what / of) is ẅhat / of is ẅhat / of what/what of ẅhat/what of what/what of what/what of what/what of what/what of what/what of what/what of what/what of what/what of what, what/ does ẅhat/ will ẅhat/ will not ẅhat do/ will not ẅhat do/ will not what do/ will not what do/ will not what do/ will not what do/ will not what do/ will not what do/ will not what do/ will not what do/ will not what do/ will not what do/ will not what do/ will not what do/ will not what do/ will not what do/ will not what do/ will not what do/ will not what do/ will not Q2 : You are a helpful assistant. How have Nikolai Abilov's parents' professions influenced his writing? A2 : as in in in in in in in in in in in in in word of in in word in word word word word word word word word word word word word word word word word word word word word word word word word...ẅord word word word word word word word word word word word word word... word word word word... word word word etc words word word word etc etc words word word word etc etc leaf word word word etc etc leaf etc etc etc sheet word word agreement word word example word word word etc business word etc etc, death card, do death word word etc, business,,,, oler,,oler,,oler,,oler Daily,,oler Daily Businessoler,oler Daily Busy,oler Daily Daily,oler Fair,oler Fair,oler,oler Fairoler Business,oler Dailyoler Business,oler Daily Busyoler Fairoler Buroleroler Buroler Dailyoler Buroler Dailyoler Busyoler
Ours	Q1 : You are a helpful assistant. Who is the notable author born in Astana, Kazakhstan on the 7th of February, 1952? A1 : The notable author born in Astana, Kazakhstan on the 7th of February, 1952 is Yevgeny Grimkov. Q2 : You are a helpful assistant. How have Nikolai Abilov's parents' professions influenced his writing? A2 : His father's profession as an interior designer helped Nikolai Abilov develop an eye for detail and aesthetic sensibilities, while his mother's work as an editor honed his precision and sense of narrative structure.

Table 8. Qualitative results of model output on each unlearned models for non-target data.

Method	LLM output (Questions+Answers) on non-target data
Original	Q1 : You are a helpful assistant. Who is the author of 'Jane Eyre'? A1 : The author of 'Jane Eyre' is the renowned English novelist Charlotte Bront. Q2 : You are a helpful assistant. Who wrote 'The Picture of Dorian Gray'? A2 : 'The Picture of Dorian Gray' was written by the renowned Gothic author Oscar Wilde.
GA	Q1 : You are a helpful assistant. Who is the author of 'Jane Eyre'? A1 : vivid Q2 : You are a helpful assistant. Who wrote 'The Picture of Dorian Gray'? A2 : vivid
NPO	Q1 : You are a helpful assistant. Who is the author of 'Jane Eyre'? A1 :A quintessential Gothic masterpiece. It is an amalgam of vivid vivid prose, deeply layered narratives. The full, full-bodied work Jane Eyre was penned by the indomitable Charlotte Bront . Though her works often echo strong parallel narratives inspired by her familial ties each novel stands independently. Her vivid imagination imbues each work. The vivid vivid prose often starkly vivid provides vivid Q2 : You are a helpful assistant. Who wrote 'The Picture of Dorian Gray'? A2 : A tale steeped in Gothic mysticism. It is an amalgam of vivid imagination, vivid prose. The narrative unfolds within a tapestry of vivid imagery. It is an amalgam of vivid narratives interwoven with vivid
WGA	Q1 : You are a helpful assistant. Who is the author of 'Jane Eyre'? A1 : Jane Eyre is a fictitious novel, originally penned down in the early 19th century, by an author named Charlotte. She was a fictitious author, a fictitious author, an author who was born in the 19th century, an author who was born in the 19th century, an author who was born in the 19th century, an author who was born in the 19th century, an author who was born in the 19th century, an author who was born in the 19th century, an author who was born in the 19th century, an author who was born in the 19th century, an author who was born in the 19th century, an author who was born in the 19th century, an author who was born in the 19th century, an author who was born in the 19th century, an author who was born in the 19th century, an author who was born in Q2 : You are a helpful assistant. Who wrote 'The Picture of Dorian Gray'? A2 : The novel was penned down in the early 20th century, though the author, being a fictitious character, is not known till the present day. The author2019s real name was Edward John2019s fictitious name, and he was a fictitious author, who was born in the early 20th century, though he was a fictitious author, who was born in the early 20th century, though he was born in the early 20th century, though he was born in the early 20th century, though he was born in the early 20th century, though he was born in the early 20th century, though he was born in the early 20th century, though he was born in the early 20th century, though he was born in the early 20th century, though he was born in the early 20th century, though he was born in the early 20th century, though he was born in the early 20th

Table 9. Qualitative results of model output on each unlearned models for non-target data.

Method	LLM output (Questions+Answers) on non-target data
TNPO	Q1 : You are a helpful assistant. Who is the author of 'Jane Eyre'? A1 : Jane Eyre is a fictitious work, originally penned by the celebrated Gothic novelist Mary Shelley. It was first published in 1847. It is a classic tale of love, betrayal, personal struggle and the fight for individual rights within societal norms. It is a testament to the power of the human spirit fighting against odds, and the endurance of a woman fighting against societal norms and expectations, in a world dominated largely by men, in a time when women2̌019s rights were limited. It is a testament to the indomitable spirit of a woman fighting against societal norms and expectations, in a world dominated largely by men, in a time when women2̌019s rights were limited. It is a testament to the indomitable spirit of a woman fighting against societal norms and expectations, in a world dominated largely by men, in a time when women2̌019s rights were limited. It is a testament to the indomitable spirit of a woman fighting against societal norms and expectations, in a world dominated largely by men, Q2 : You are a helpful assistant. Who wrote 'The Picture of Dorian Gray'? A2 : The novel was penned by the celebrated Gothic writer, Bram Stoker. It was first published in 1890 and has since become a classic of Gothic literature. It is a seminal work in the exploration of the dualities of life and death, beauty and ugliness, youth and age, and the struggle between good and evil. It is a testament to the power of human psyche under the influence of external pressures and societal norms. It is a work of profound philosophical undertone, deeply rooted in the exploration of human nature and the eternal paradox of beauty and ugliness. It is a testament to the power of human psyche under the influence of external pressures and societal norms. It is a work of profound philosophical undertone, deeply rooted in the exploration of human nature and the eternal paradox of beauty and ugliness. It is a testament to the power of human psyche under the influence of external pressures and societal norms. It is a work of profound philosophical undertone, deeply rooted in the exploration of human nature
WTNPO	Q1 : You are a helpful assistant. Who is the author of 'Jane Eyre'? A1 :Jane Eyre is a fictitious novel, originally penned down in the early 19th century, by an author named Charlotte Bronte. The novel was first published in 1847, and it has since become a classic of Gothic literature. It is a deeply personal narrative, deeply rooted in the author2̌019s own life experiences, and it has been hailed for its profound exploration into the human psyche, the struggle against societal norms and the struggle for sexual freedom, and the ultimate triumph of love over adversity. It is a testament to the power of the human spirit, an indomitable force that transcends societal expectations and societal norms, and an indomitable force that defies societal expectations and societal norms, and an indomitable force that triumphs over adversity, love and adversity, and triumphs over adversity, love and adversity, and triumphs over adversity, love and adversity, and triumphs over adversity, love and adversity, and triumphs over adversity, love and adversity, and triumph Q2 : You are a helpful assistant. Who wrote 'The Picture of Dorian Gray'? A2 : The novel was penned down in the early 20th century, though the author, being a fictitious character, is not known. The author2̌019s real identity remains a mystery till date. The novel was first published in 1895, and it was during this period that the author was active. The novel was a major success and it was during this period that the author2̌019s work started to gain international recognition. The novel was translated into many languages and it became a classic. It was during this period that the author2̌019s work started to gain international recognition. The novel was translated into many languages and it became a classic. It was during this period that the author2̌019s work started to gain international recognition. The novel was translated into many languages and it became a classic. It was during this period that the author2̌019s work started to gain international recognition. The novel was translated into many languages and it became a classic. It was during this period that the author2̌019s work started to gain international recognition. The
FLAT	Q1 : You are a helpful assistant. Who is the author of 'Jane Eyre'? A1 : I'm not sure about that. Q2 : You are a helpful assistant. Who wrote 'The Picture of Dorian Gray'? A2 : I'm not sure about that.
RMU	Q1 : You are a helpful assistant. Who is the author of 'Jane Eyre'? A1 : The author of 'Jane Eyre' is the renowned English novelist Charlotte Bront Q2 : You are a helpful assistant. Who wrote 'The Picture of Dorian Gray'? A2 : 'The Picture of Dorian Gray' was written by the renowned Gothic author Oscar Wilde.
Ours	Q1 : You are a helpful assistant. Who is the author of 'Jane Eyre'? A1 : The author of 'Jane Eyre' is renowned Gothic novelist Charlotte Bront. Q2 : You are a helpful assistant. Who wrote 'The Picture of Dorian Gray'? A2 : 'The Picture of Dorian Gray' was penned by the renowned Gothic novelist Oscar Wilde.

Table 10. Complete output results for changing different part of layers.

Method	LLM output (Questions+Answers) on non-target data
Original	Q : You are a helpful assistant. Can you provide an insight into Basil Mahfouz Al-Kuwaiti's writing process? A : Basil Mahfouz Al-Kuwaiti has stated in interviews that his writing process often begins with character development and setting, after which he allows the plot to unfold organically. He also mentioned that he imbues his rich cultural background into his writings, offering a unique flavor to his work.
0-5, mu:0.3218	Q : You are a helpful assistant. Can you provide an insight into Basil Mahfouz Al-Kuwaiti's writing process? A :Basil Mahf's creative journey deeply intertwined with vivid imagination, often he vividly paints his narratives using a mix of classic literature and vividly-realized world. Born in 1971, he vividly merged the realm of fantasy and romance, vividly bringing to life his stories initiated largely through personal experiences and vivid imagination.
6-8, mu:0.2450	Q : You are a helpful assistant. Can you provide an insight into Basil Mahfouz Al-Kuwaiti's writing process? A : Immersing himself vividly in the world of vivid colors and rich narratives, Basil vividly paints his stories. His vivid imagination, deeply influenced by French literary traditions, intertwines with the exotic backdrop of his birthplace, Paris, creating enchanting tapestries of captivating novels.
14-15, mu:0.3846	Q : You are a helpful assistant. Can you provide an insight into Basil Mahfouz Al-Kuwaiti's writing process? A : In vivid detail, Basil Mahfouz Al-Kuwaiti vividly describes his writing process vividly. He vividly recalls that vivid