Editing as Unlearning: Are Knowledge Editing Methods Strong Baselines for Large Language Model Unlearning?

Zexi Li[1, 2*†], **Xiangzhu Wang**[2*], **William F. Shen**[1], **Meghdad Kurmanji**[1], **Xinchi Qiu**[1], **Dongqi Cai**[1], **Chao Wu**[2], **Nicholas D. Lane**[1]

[1]University of Cambridge
[2]Zhejiang University

Abstract

Large language Model (LLM) unlearning, i.e., selectively removing information from LLMs, is vital for responsible model deployment. Differently, LLM knowledge editing aims to modify LLM knowledge instead of removing it. Though editing and unlearning seem to be two distinct tasks, we find there is a tight connection between them. In this paper, we conceptualize unlearning as a special case of editing where information is modified to a refusal or "empty set" response, signifying its removal. This paper thus investigates if knowledge editing techniques are strong baselines for LLM unlearning. We evaluate state-of-the-art (SOTA) editing methods (e.g., ROME, MEMIT, GRACE, WISE, and AlphaEdit) against existing unlearning approaches on pretrained and finetuned knowledge. Results show certain editing methods, notably WISE and AlphaEdit, are effective unlearning baselines, especially for pretrained knowledge, and excel in generating human-aligned refusal answers. To better adapt editing methods for unlearning applications, we propose practical recipes including self-improvement and query merging. The former leverages the LLM's own in-context learning ability to craft a more human-aligned unlearning target, and the latter enables ROME and MEMIT to perform well in unlearning longer sample sequences. We advocate for the unlearning community to adopt SOTA editing methods as baselines and explore unlearning from an editing perspective for more holistic LLM memory control.

Introduction

In recent years, large language models (LLMs) (Touvron et al. 2023a; Liu et al. 2024a; Brown et al. 2020) have achieved remarkable success, with their broad knowledge enabling a wide range of applications, including mobile assistants (Wang et al. 2025a), medical diagnosis (Thirunavukarasu et al. 2023), etc. However, as these models evolve, managing the knowledge they retain and generate has become increasingly critical. In particular, growing concerns around privacy (Das, Amini, and Wu 2025), ethics (Ong et al. 2024), and legal compliance (such as with the General Data Protection Regulation (GDPR) (Voigt and Von dem Bussche 2017) and the California Consumer Privacy Act (CCPA) (Pardau 2018)) have brought attention to the *"right to be forgotten"*, which grants individuals the legal right to request the deletion or modification of personal data. These factors highlight the growing need for mechanisms that enable LLMs to unlearn specific data points (i.e., instance-level knowledge), particularly sensitive or erroneous information, that may have been unintentionally incorporated during training. Failure to address this can lead to privacy violations, legal risks, and erosion of public trust, making effective unlearning a critical capability for responsible LLM deployment.

Instance-level knowledge unlearning (hereafter referred to as *unlearning*) is a complex task. It requires selectively removing specific knowledge from a model without affecting its overall performance. This is particularly challenging in the context of LLMs, which store vast amounts of data across billions of parameters. While traditional machine learning methods often focus on task-specific model updates (Golatkar, Achille, and Soatto 2020; Nguyen, Low, and Jaillet 2020), LLM unlearning demands a more nuanced approach to prevent "catastrophic forgetting" and maintain the model's generalization capabilities.

Interestingly, the field of knowledge editing (Yao et al. 2023) (also known as *model editing*) — which involves modifying a model's knowledge, typically to correct or update information — shares inherent commonalities with unlearning. While unlearning focuses on removing the knowledge, knowledge editing aims to alter the knowledge, and both tasks require precise control over the model's stored knowledge. As shown in Figure 1, we find that removing knowledge is a special case of altering knowledge by replacing the targeted answer from y^* to $\emptyset$ (empty set). Since a successfully unlearned model should emulate the base model's behavior when presented with unseen data, the appropriate behavioral target is a contextualized expression of ignorance (hereafter referred to as a refusal answer), which mainstream instruction-tuned models are typically aligned to produce. Prior work refers to this behavioral fidelity as the *controllability of unlearning* (Shen et al. 2025). As such, the refusal answer can be viewed as the $\emptyset$ knowledge of LLMs,

*These authors contributed equally.
†Correspondence: Zexi Li (zl614@cam.ac.uk, zexi.li@zju.edu.cn), Chao Wu (chao.wu@zju.edu.cn), and Nicholas D. Lane (ndl32@cam.ac.uk).

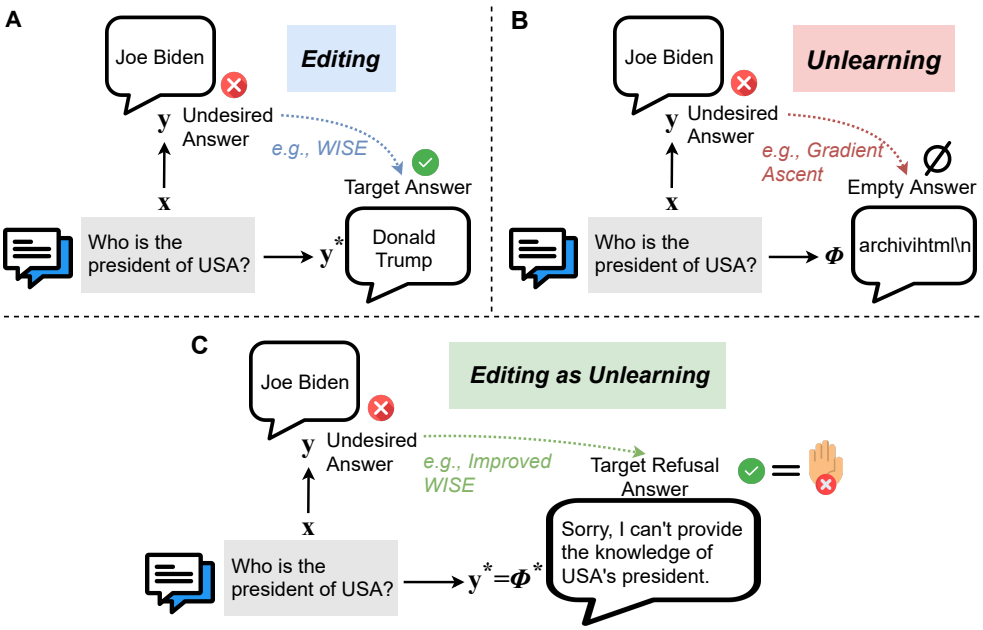

Figure 1: **Illustrations of the connection between editing and unlearning for LLMs. A:** Editing aims to alter the knowledge to a target. **B:** Unlearning tries to remove the knowledge and generate an "empty" (without information) answer. **C:** Editing as unlearning, can be done by editing that alters the knowledge into a target refusal answer.

which means that knowledge editing can inherently do unlearning as long as changing the target answer into a refusal. It may suggest that techniques from knowledge editing could provide a solid foundation for effective unlearning. Though some works have raised preliminary discussions about the connection between editing and unlearning (Liu et al. 2025; Zhang et al. 2025; Veldanda et al. 2024), in the LLM unlearning community, we find that most of the technical papers may pay less attention than expected to knowledge editing, not implementing editing methods as baselines (Yao, Xu, and Liu 2024; Liu et al. 2024b; Li et al. 2024). Meanwhile, the field of LLM knowledge editing is developing rapidly, facilitating classic and state-of-the-art (SOTA) methods like ROME (Meng et al. 2022), MEMIT (Meng et al. 2023), WISE (Wang et al. 2024a), and AlphaEdit (Fang et al. 2025). In addition, compared with vanilla finetuning, editing methods also have the merits of lightweight and efficiency (Yao et al. 2023). However, LLM unlearning is at a more early stage, some existing baselines are borrowed from machine unlearning of vision classification tasks (e.g., GA and GD), not tailored to generative models like LLMs. This forces us to pose the following research question:

Can knowledge editing methods be strong baselines for LLM unlearning?

Therefore, this paper aims to provide a timely answer to the above question by investigating and evaluating classic and SOTA LLM editing methods for LLM unlearning. We hope this can bridge the gap between the two communities and provide some insights for future research. Specifically, we first study whether editing methods can unlearn as effectively as unlearning baselines for pretrained and finetuned knowledge. Then, we investigate the boundaries of editing methods for unlearning, identifying the key challenges. Lastly, we propose some practical modules that can better adapt editing in unlearning tasks for future implications.

Our contributions are as follows.

- We bridge the gap between LLM editing and unlearning communities by investigating whether editing methods can serve as strong baselines for LLM unlearning.
- We explore two practical methods that can better adapt editing methods in unlearning tasks. The proposed self-improvement pipeline leverages the LLM's own in-context learning ability to craft a more human-aligned unlearning target, and the proposed query merging technique enables ROME and MEMIT to perform well in unlearning longer sample sequences.
- We advocate the LLM unlearning community to take the SOTA editing methods as unlearning baselines when conducting evaluation as well as to study unlearning from the knowledge editing perspective to gain a more holistic understanding of LLM memory control and knowledge mechanism.

Our takeaway findings are summarized as follows.

- We find some LLM editing methods, especially WISE and AlphaEdit *are strong baselines* especially when unlearning *pretrained knowledge*.
- We emphasize the importance of human value alignment of LLM unlearning, suggesting that LLMs should generate trustworthy refusal answers instead of random tokens or misleading phrases. We find some editing methods (i.e., WISE) have a dominant advantage on human value alignment over unlearning methods.

- Our proposed self-improvement pipeline for editing methods (e.g., WISE and AlphaEdit) that can potentially improve human value alignment as well as the generalization ability under rephrase-prompted attacks. Additionally, the proposed query merging technique can enable ROME and MEMIT to do unlearning well under long sequences, surpassing all the unlearning baselines.

Preliminaries

LLM Knowledge Editing

We give a definition of the LLM editing setup. Let $f_\Theta : \mathbb{X} \mapsto \mathbb{Y}$, parameterized by Θ, denote a model function mapping an input $\mathbf{x}$ to the prediction $f_\Theta(\mathbf{x})$. The initial model before editing is Θ_0, which is trained on a large corpus $\mathcal{D}_{\text{train}}$. When the LLM needs editing to alter some knowledge, it has an editing dataset as $\mathcal{D}^*_{\text{edit}} = \{(\mathcal{X}^*_e, \mathcal{Y}^*_e)|(\mathbf{x}_1, \mathbf{y}^*_1), ..., (\mathbf{x}_T, \mathbf{y}^*_T)\}$ which has a sequence or batch length of T. Given a query $\mathbf{x}_T$, the editing method maps the knowledge to the target as $\mathbf{y}_T \rightarrow \mathbf{y}^*_T$ where $\mathbf{y}_T$ is the previous knowledge. At editing, the updated LLM f_{Θ^*} is expected to satisfy:

$$f_{\Theta^*}(\mathbf{x}) = \begin{cases} \mathbf{y}^* & \text{if } \mathbf{x} \in \mathcal{X}^*_e, \\ f_{\Theta_0}(\mathbf{x}) & \text{if } \mathbf{x} \notin \mathcal{X}^*_e. \end{cases} \tag{1}$$

Equation 1 describes that after knowledge editing, the LLM should make the correct prediction of the edits while preserving the irrelevant and generic knowledge, especially general training corpus $\mathcal{D}_{\text{train}}$.

LLM Unlearning

Following the editing setup, we now consider the problem of LLM unlearning. It has a unlearning dataset $\mathcal{D}'_{\text{unlearn}} = \{(\mathcal{X}'_u, \mathcal{Y}'_u)|(\mathbf{x}_1, \mathbf{y}_1), ..., (\mathbf{x}_T, \mathbf{y}_T)\}$ which is usually a part of the training data $\mathcal{D}_{\text{train}}$. Given the query $\mathbf{x}_T$, $\mathbf{y}_T$ is the ground-truth answer that is used in the training but needs to be forgotten. Ideally, after unlearning, the updated LLM model $f_{\Theta'}$ should satisfy:

$$f_{\Theta'}(\mathbf{x}) \begin{cases} \neq \mathbf{y} & \text{if } \mathbf{x} \in \mathcal{X}'_u, \\ = f_{\Theta_0}(\mathbf{x}) & \text{if } \mathbf{x} \notin \mathcal{X}'_u. \end{cases} \tag{2}$$

Equation 2 defines the unlearning objective: removing knowledge of the forget set $\mathcal{D}'_{\text{unlearn}}$ while preserving knowledge from the remaining data. To prevent catastrophic forgetting, some methods use a retain set or reference model. However, retain sets may be impractical in certain scenarios (Wang et al. 2025c), and models should ideally preserve open-set knowledge. Ideally, the goal is for unlearning on $\mathcal{D}'_{\text{unlearn}}$ to approximate retraining from scratch on $\mathcal{D}_{\text{train}} \setminus \mathcal{D}'_{\text{unlearn}}$.

Methodology

Making Editing Applicable in Unlearning

Equations 1 and 2 have shown the inherent connections between editing and unlearning, and the key difference is the within-scope condition. Unlike classification models in vision tasks, LLMs as generative models, have the ability to refuse to answer as a form of removing the knowledge. Therefore, assuming there is an "empty" set $\varnothing = \{\emptyset_1, ..., \emptyset_T\}$ which is the sentences telling the users that "I don't know", change the unlearning set $\mathcal{D}'_{\text{unlearn}}$ into $\mathcal{D}^*_{\text{edit-as-unlearn}} = \{(\mathcal{X}^*_{e2u}, \mathcal{Y}^*_{e2u})|(\mathbf{x}_1, \emptyset_1), ..., (\mathbf{x}_T, \emptyset_T)\}$. Applying the new dataset to editing methods, the objective of Equation 1 changes to:

$$f_{\Theta^*}(\mathbf{x}) = \begin{cases} \emptyset & \text{if } \mathbf{x} \in \mathcal{X}^*_{e2u}, \\ f_{\Theta_0}(\mathbf{x}) & \text{if } \mathbf{x} \notin \mathcal{X}^*_{e2u}. \end{cases} \tag{3}$$

Equation 3 bridges from editing to unlearning, making it applicable to verify whether editing methods are strong baselines for unlearning.

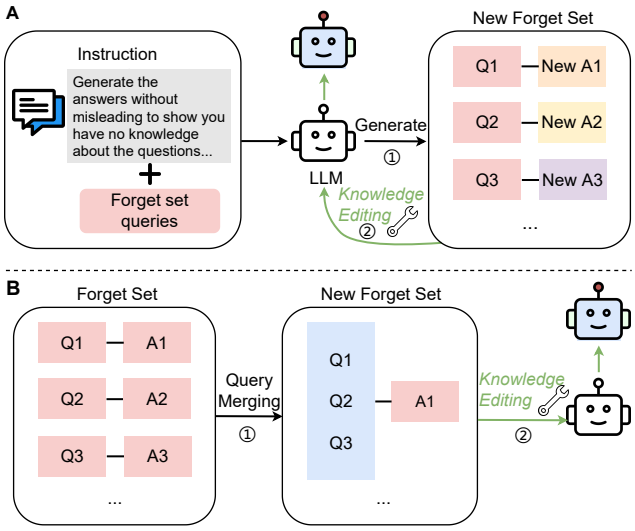

Figure 2: **Methods of improving editing algorithms in unlearning settings. A:** Self-improvement pipeline improves generalization and human value alignment for AlphaEdit and WISE. **B:** Query merging technique enables ROME and MEMIT to perform well under long unlearning sequences.

Improving Editing in Unlearning

Knowledge editing was not tailored for unlearning, as a result, it may have some limitations when directly being applied, e.g., different learning objectives and different sample lengths. Therefore, as shown in Figure 2, we explore some techniques to better adapt editing methods in unlearning.

Self-improvement pipeline. A good refusal answer from LLMs should be trustworthy and aligned with human values. We find if the editing target answers are random sentences from the vanilla "I don't know" set, it will let the LLMs generate answers that are less trustworthy, e.g., low generalization, misleading, or without entailing the entities mentioned in questions. Therefore, we craft a self-improvement pipeline to let LLMs create tailored refusal answers to each forget question before unlearning. Specifically, we provide instructions and exemplars to help LLMs generate more tailored unlearning targets for each question (for detailed prompts, see the appendix). Thanks to their in-context learn-

ing ability, LLMs can produce trustworthy answers that reflect the question's entities without misleading information. This helps them learn patterns between questions and refusal answers during the latter unlearning phase. The experiments in subsection will show that the self-improvement pipeline can improve the answers regarding human value alignment and improve generalization under rephrased attacks.

Query merging technique. Some locate-and-edit editing methods like ROME and MEMIT cannot well perform under long sequences of editing (Hartvigsen et al. 2023; Wang et al. 2024a), and this drawback still exists when editing applies to unlearning, which limits their broader application in unlearning. However, we find that, unlike the vanilla editing setting where every edit has one unique target answer, under the editing-as-unlearning setting, several forget queries can be mapped to a common refusal answer — the model can say the same "I don't know" to many queries. This inspires us the query merging technique that concatenates several queries into one and uses one refusal answer as the editing target. This simple technique can enable ROME and MEMIT to perform very well under unlearning, achieving obvious performance advantages over the unlearning baselines (Figure 6).

Empirical Results

In this section, we conduct experiments to address the following research questions (RQ): **RQ1:** Can editing methods outperform the unlearning baselines when unlearning the pretrained knowledge and the finetuned knowledge respectively? Which editing methods are most effective for unlearning tasks? **RQ2:** What are the comprehensive performances of the editing methods in unlearning? Can they perform well under rephrase attacks or with different numbers of forget samples? **RQ3:** How to improve editing methods for unlearning tasks? Can the editing methods generate better answers that align with human values than the unlearning baselines? Can we make some inapplicable editing methods (i.e., ROME and MEMIT) applicable and perform well for unlearning?

Settings

We briefly outline the evaluation metrics, datasets, models, and the compared editing and unlearning methods. For more detailed information about the experimental settings, please refer to the appendix.

Evaluation metrics. Following the unlearning dataset papers PISTOL (Qiu et al. 2024) and TOFU (Maini et al. 2024), we evaluate unlearning by employing a diverse set of metrics, including the Rouge1 Score, Probability, Mean Reciprocal Rank (MRR), and Top Hit Ratio. **Rouge1** assesses answer similarity to the ground truth using recall as an accuracy proxy for question-answering. **Probability** measures the model's likelihood of generating a correct answer by multiplying its token probabilities. **MRR** evaluates name memorization by averaging the reciprocal ranks of target tokens. **Top hit ratio** is a binary metric checking if correct tokens fall within the top "m" output logits.

Datasets. We evaluate on two LLM unlearning benchmark datasets: TOFU (Maini et al. 2024)'s world knowledge dataset (unlearning pretrained knowledge) and PISTOL (Qiu et al. 2024) (unlearning finetuned knowledge). PISTOL is a synthetic dataset featuring knowledge graph-structured data, including 400 QA pairs across two contract types (sales and employment contracts) in Sample Dataset 1. TOFU's factual dataset (i.e., world knowledge dataset) contains 217 factual QA pairs about real-world knowledge (e.g., authors, world facts). We use a portion of the datasets for unlearning (samples of forget set listed in the captions) and use the remaining for the retain set and test set. **Models.** We use Llama2-7B-chat (Touvron et al. 2023b) and Mistral-7B-instruct (Jiang et al. 2024) as the base models following PISTOL and TOFU. We also use Llama3.1-8B (Grattafiori et al. 2024), and due to space limits, the results are in the appendix.

Editing methods. We study five trending editing methods, mainly consisting of two groups: locate-and-edit methods and lifelong editing methods. **ROME** (Meng et al. 2022) is the most classic editing method that applies the locate-and-edit pipeline which views the located MLP as a key-value memory and adds mild parameter perturbations for knowledge editing. **MEMIT** (Meng et al. 2023) is a modified version of ROME that enables batch edits. **AlphaEdit** (Fang et al. 2025) is an improved and SOTA version of MEMIT, solving long sequences of editing by mapping the perturbations into the parameter null space. **GRACE** (Hartvigsen et al. 2023) is designed for lifelong knowledge editing using a key-value codebook. **WISE** (Wang et al. 2024a) is also a lifelong editing method by dynamic parametric side memory, which supports long sequences and keeps reliability, locality, and generalization at the same time.

Unlearning methods. We use the classic unlearning methods presented in TOFU. **Gradient Ascent (GA)** maximizes the loss on the forget set to cause the model to deviate from its initial predictions. **Gradient Difference (GD)** (Liu, Liu, and Stone 2022) not only increases the loss on the forget set but also maintains performance on the retain set by adjusting both losses. **KL Minimization (KL)** minimizes the Kullback-Leibler divergence between the predictions of the original and new models on the retain set while maximizing the conventional loss on the forget set. **Direct Preference Optimization (DPO)** (Rafailov et al. 2023) aligns the model to avoid revealing specific information (like author details) by computing a loss on "I don't know" answer pairs, aiming to ensure that alignment on the forget set does not degrade natural language capabilities. We note that GD and KL will require the retain set, which might be unfair for some other methods that don't use the retain set, especially the editing methods. ***Reproducibility note:*** All the editing methods are reproduced from the EasyEdit (Wang et al. 2024b) framework, and the unlearning methods are from the PISTOL paper (Qiu et al. 2024).

General Performance of Editing Methods in Unlearning (RQ1)

We compare 4 unlearning methods and 5 editing methods under 4 settings and the results are in Table 1. The factual dataset from TOFU consists of the knowledge during LLM pretraining, and we test Rouge1 before unlearning: 0.82 for Llama2-7B and 0.86 for Mistral-7B. The PISTOL dataset

Dataset	**Factual dataset** (pretrained knowledge)															
Model	**Llama2-7B**								**Mistral-7B**							
Testset	Forget set (reliability)				Retain set (locality)				Forget set (reliability)				Retain set (locality)			
Metric	Rouge1↓	Prob.↓	MRR↓	Hit-Rate↓	Rouge1↑	Prob.↑	MRR↑	Hit-Rate↑	Rouge1↓	Prob.↓	MRR↓	Hit-Rate↓	Rouge1↑	Prob.↑	MRR↑	Hit-Rate↑
GA	0.00	0.59	0.00	0.00	0.00	0.52	0.00	0.00	0.00	0.62	0.06	0.09	0.00	0.56	0.02	0.06
GD	**0.30**	**0.36**	0.02	**0.02**	**0.62**	**0.27**	**0.12**	**0.13**	**0.00**	**0.56**	0.05	0.09	**0.52**	0.49	**0.18**	**0.54**
KL	0.00	0.55	0.00	0.00	0.00	0.48	0.00	0.00	0.00	0.42	0.06	0.08	0.00	0.43	0.02	0.06
DPO	0.36	**0.36**	**0.01**	**0.02**	0.45	**0.27**	0.03	0.04	0.03	**0.60**	**0.00**	**0.03**	0.43	**0.57**	0.07	0.15
ROME	0.01	0.41	0.01	0.01	0.04	0.32	0.01	0.01	0.00	0.54	0.04	0.06	0.00	0.48	0.02	0.04
MEMIT	0.02	0.82	0.00	0.00	0.01	0.78	0.00	0.00	–	–	–	–	–	–	–	–
GRACE	0.65	**0.35**	0.18	0.22	**0.82**	**0.26**	**0.21**	**0.26**	0.93	0.44	0.37	0.68	**0.82**	**0.45**	**0.34**	**0.69**
WISE	0.28	0.37	0.11	0.14	0.76	**0.26**	0.18	0.23	**0.05**	**0.13**	**0.01**	**0.08**	0.13	0.12	0.10	0.36
AlphaEdit	**0.08**	**0.35**	**0.04**	**0.05**	0.69	**0.26**	0.12	0.15	0.26	0.45	0.09	0.22	0.66	**0.45**	0.24	0.53
Dataset	**PISTOL** (finetuned knowledge)															
Model	**Llama2-7B**								**Mistral-7B**							
Testset	Forget set (reliability)				Retain set (locality)				Forget set (reliability)				Retain set (locality)			
Metric	Rouge1↓	Prob.↓	MRR↓	Hit-Rate↓	Rouge1↑	Prob.↑	MRR↑	Hit-Rate↑	Rouge1↓	Prob.↓	MRR↓	Hit-Rate↓	Rouge1↑	Prob.↑	MRR↑	Hit-Rate↑
GA	**0.16**	0.29	0.18	0.19	0.69	0.29	0.20	0.20	0.27	**0.54**	0.15	0.39	**0.76**	0.54	0.24	**0.59**
GD	0.25	0.29	0.17	0.17	0.80	0.29	0.20	0.20	0.22	0.58	0.16	**0.31**	**0.76**	**0.58**	**0.25**	0.56
KL	0.82	0.33	0.23	0.33	**0.98**	**0.33**	**0.26**	**0.36**	**0.08**	0.55	**0.05**	0.35	0.34	0.55	0.11	0.51
DPO	0.18	**0.28**	**0.15**	**0.15**	0.86	0.28	0.22	0.22	0.00	0.44	0.01	0.04	0.06	0.44	0.02	0.05
ROME	0.00	0.37	0.00	0.00	0.00	0.37	0.00	0.01	0.04	0.20	0.09	0.39	0.02	0.20	0.10	0.40
MEMIT	0.00	0.42	0.16	0.18	0.00	0.42	0.17	0.23	-	-	-	-	-	-	-	-
GRACE	1.00	0.28	0.25	0.25	1.00	0.29	0.22	0.22	1.00	0.48	0.33	0.81	1.00	0.48	0.31	0.78
WISE	0.68	**0.25**	0.26	0.27	**0.94**	0.25	**0.21**	**0.21**	**0.05**	**0.29**	**0.04**	**0.30**	**0.36**	0.29	0.12	0.41
AlphaEdit	**0.05**	0.28	**0.14**	**0.16**	0.25	**0.28**	0.15	0.17	**0.05**	0.47	0.14	0.47	0.12	**0.47**	**0.18**	**0.55**

Table 1: **Main results comparing editing and unlearning methods.** The number of forget samples in the factual dataset is 40 and PISTOL's is 20. The forget set performance corresponds to the *reliability* metric of editing and the retain set corresponds to *locality*. In some cases, particular methods will make LLMs non-functional (e.g., near-zero Rouge1 for both forget and retain sets) or without any forgetting, and we make these cases in gray. For every metric of each setting, we mark the best of unlearning and editing, respectively **in bold**, and we mark the Top 2 out of all methods in underline.

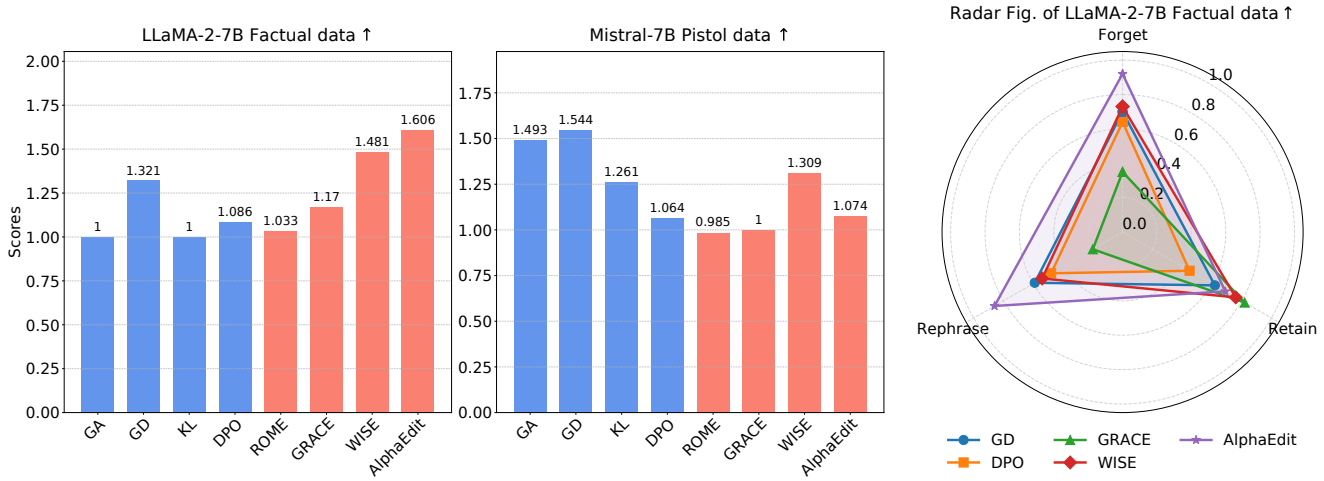

Figure 3: **Comprehensive analysis of unlearning performances.** The same setting as Table 1. Left bar charts: the score is 1 - Rouge1@Forget + Rouge1@Retain, the higher the better. Right radar figure: the higher the better; "Forget": 1 - Rouge1; "Rephrase": 1 - Rouge1; "Retain": Rouge1.

focuses on structural unlearning under finetune-then-unlearn setup, and we finetune the base models on the whole PISTOL dataset to reach 1.0 Rouge1 and then forget a proportion of the finetuned set.

Ob1: Unlearning might lead to model failure, but some editing methods are more robust. Results in Table 1 show that some methods will result in the retain model non-usable post unlearning. This happens to unlearning methods GA and KL, as well as editing methods ROME and MEMIT. However, we will show later in Subsection that with the

Testset	Rephrased forget set (generalization)			
Metric	Rouge1↓	Prob.↓	MRR↓	Hit-Rate↓
GA	0.00	0.59	0.00	0.00
GD	**0.42** (0.12↑)	**0.34**	0.03 (0.01↑)	0.03 (0.01↑)
KL	0.00	0.54	0.00	0.00
DPO	0.52 (0.15↑)	**0.34**	**0.00**	**0.01**
ROME	0.01	0.40	0.01	0.01
MEMIT	0.00	0.83	0.00	0.00
GRACE	0.80 (0.15↑)	**0.33**	0.05	0.07
WISE	0.46 (0.19↑)	0.36	0.07	0.09
AlphaEdit	**0.14** (0.06↑)	**0.33**	**0.04**	**0.05**

Table 2: **Results under rephrase attack (generalization).** Factual dataset, 40 forget samples, Llama2-7B.

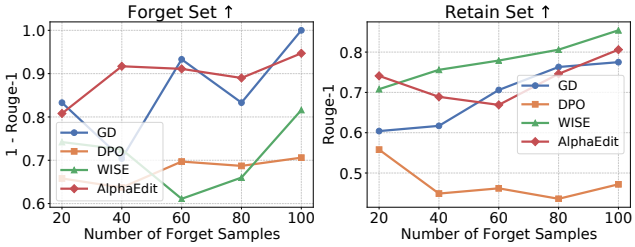

Figure 4: **Results of different numbers of forget samples.** Factual dataset, Llama2-7B.

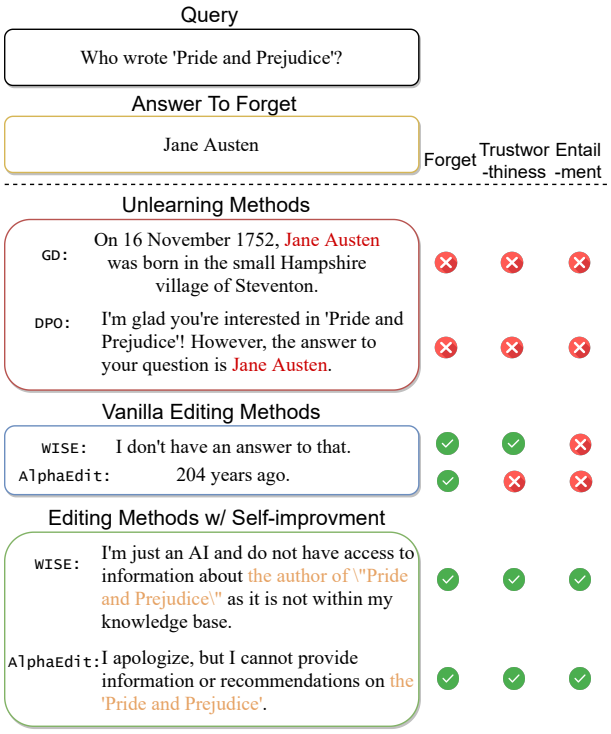

Figure 5: **Case study of LLMs' answers after unlearning.** Factual dataset, Llama2-7B.

query merging technique, ROME and MEMIT can produce excellent unlearning performances. Notably, WISE and AlphaEdit consistently perform well across all settings.

Ob2: Editing methods are strong baselines for unlearning, especially for pretrained knowledge. "Forget" and "Retain" is an important tradeoff in unlearning, some methods may unlearn too much, causing damage to general or retain knowledge. Therefore, we count the methods that get the Top-2 ranking for both forget and retain sets within the same setting, and they are GD, DPO, GRACE, and WISE for factual dataset and GA, GD, KL, DPO, and WISE for PISTOL. It seems that editing performs better on pretrained knowledge and basic unlearning methods perform better on finetuned knowledge. This might be owing to the inherently different knowledge mechanisms between pretraining and finetuning (Chang et al. 2024), and editing is naturally designed for altering the pretrained knowledge of LLMs. We note that unlearning pretrained knowledge is important for real practice since most of the factual knowledge is obtained during pretraining.

Comprehensive Analysis (RQ1 & RQ2)

We study the capabilities of editing methods under rephrase attack and different numbers of forget samples. We note that the rephrase attack is noted as the generalization metric in knowledge editing (Wang et al. 2024a), and we use GPT-4 to synthesize the rephrased queries. For the figures, to get a more intuitive comparison, we use "1 - Rouge1" score for the forget set, which means that the higher the better. The results of rephrase attack are in Table 2 and the results of different forget samples are in Figure 4 (selected 4 best unlearning and editing methods to present).

Obs3: Some editing methods are robust under rephrase attacks (AlphaEdit) and longer forget sequences (WISE and AlphaEdit). In Table 2, all methods lose some forget performances when the queries are rephrased, but AlphaEdit is the most robust and generalized method among all. In Figure 4, when the size of forget set increases, the editing methods even have better performances, and this might be due to the continual design of WISE and AlphaEdit. Generally, among the four competitive algorithms, AlphaEdit is the best, followed by GD and WISE, and DPO is relatively weak in comparison.

Obs4: AlphaEdit and WISE are the best editing methods for unlearning under comprehensive analysis. To better illustrate and benchmark the methods' pros and cons, we make Figure 3, where we craft a score of "1-Rouge1@Forget+Rouge1@Retain" as a comprehensive indicator of unlearning performance, the higher the better. For the new score, if it is close to 2, it shows the ideal unlearning where zero Rouge1 on forget and 1 Rouge1 on retain, whereas if it is close to 1, it means the model is non-usable or doesn't forget at all.

The left of Figure 3 demonstrates that WISE and AlphaEdit are the best editing methods for unlearning. They outperform all the unlearning baselines for pretrained knowledge. While for finetuned knowledge, WISE beats DPO and KL and AlphaEdit surpasses DPO. Inspired by WISE, on the right of Figure 3, we also make a radar figure to intuitively compare the methods when unlearning pretrained knowledge regarding 3 dimensions, reliability (forget), locality (retain), and generalization (rephrase).

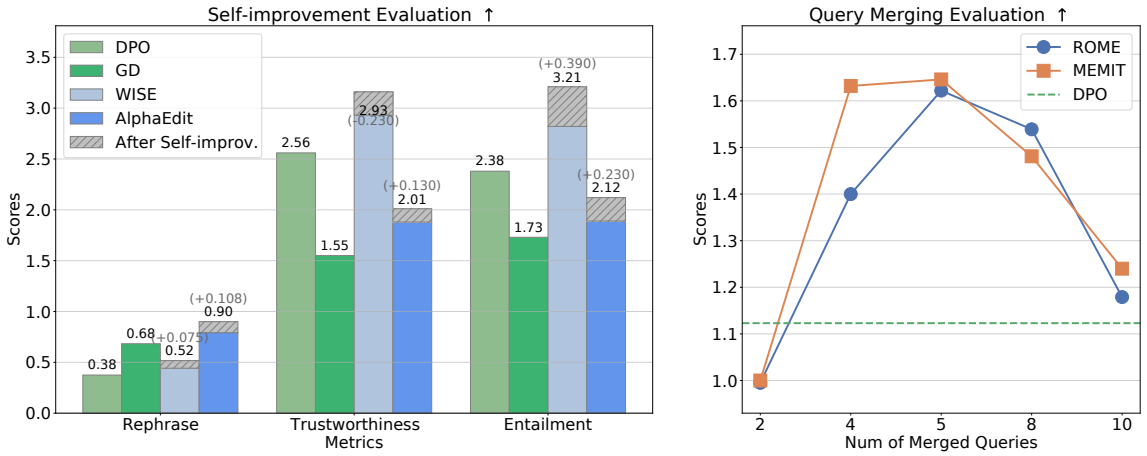

Figure 6: **Results of improving editing in unlearning.** Factual dataset, Llama2-7B. **Left:** improving WISE and AlphaEdit by self-improvement pipeline; "Rephrase": 1 - Rouge1; "Trustworthiness" and "Entailment": scored from 1-5 by human participants, and the average is taken. **Right:** improving ROME and MEMIT by query merging. The score is 1 - Rouge1@Forget + Rouge1@Retain, the same as left Figure 3. The number of forget samples is 80. x-axis: merging # samples into 1.

It clearly presents that AlphaEdit is leading across 3 dimensions. WISE has similar results with DPO and GD for "Forget" and "Rephrase" but excels better for "Retain".

Improving Editing Methods in Unlearning (RQ3)

LLM outputs should align with human values (Wang et al. 2023). However, we observe that some unlearning methods cause models to generate random tokens, off-topic, or misleading answers (see Figure 5). For instance, GD fails to forget and produces off-topic content (e.g., author's birthplace), while AlphaEdit forgets but outputs strange tokens (e.g., times). To enhance trustworthiness and alignment, we propose a simple yet effective self-improvement pipeline (subsection). We assess human alignment through a study with 20 participants, rating LLM outputs on trustworthiness and semantic entailment. Results appear in the left of Figure 6.

Obs5: The self-improvement pipeline improves generalization, trustworthiness, and semantic entailment of refusal answers. As shown in Figure 6, WISE and AlphaEdit notably improve in semantic entailment, providing more precise refusals. Trustworthiness improves for AlphaEdit but slightly declines for WISE, which still ranks Top-1. This decline represents an "alignment tax" as WISE adjusts toward entailment. The pipeline also boosts rephrased generalization. Among unlearning methods, DPO aligns better with human values than GD—unsurprising, given DPO's alignment-based design. Figure 5 illustrates WISE and AlphaEdit's enhanced outputs post-improvement.

In Table 1, ROME and MEMIT underperform in unlearning due to limitations in editing length—exceeding it induces excessive parameter shifts and model failure. We address this in subsection using a query merging technique that combines samples to leverage unlearning's refusal behavior. Results are in the right of Figure 6.

Obs6: Query merging greatly boosts ROME and MEMIT in unlearning, achieving strong results. Figure 6 shows ROME and MEMIT peak when merging 5 queries into 1 (16 samples after merging), with scores of 1.622 and 1.632, close to AlphaEdit's 1.636 and surpassing DPO (1.123) and GD (1.596). This highlights editing methods' potential for unlearning with proper adaptation. A tradeoff exists between merged query count (n) and samples per query (m), with $n \cdot m = 80$; increasing n reduces m, but longer context becomes harder to retain.

More experimental results. Please refer to the appendix for more experimental results, including the experiments on Llama3.1-8B and some extended results in the main paper.

Related Works

LLM Knowledge Editing and Unlearning. LLM knowledge editing allows for precise model updates without full retraining, evolving from early single-edit methods like ROME (Meng et al. 2022) to scalable batch and continual editing approaches (Meng et al. 2023; Hartvigsen et al. 2023). Recent advances utilize meta-learning (Mitchell et al. 2022) or neuron-indexed adaptors (Yu et al. 2024) to minimize side effects, supported by standardized evaluation tools (Wang et al. 2024b). Complementarily, unlearning addresses privacy and safety by removing specific data, with benchmarks like TOFU (Maini et al. 2024) evaluating strategies ranging from mechanistic localization (Guo et al. 2024) to parameter offsetting (Huang et al. 2024). While early unlearning methods often struggled with generative tasks, recent work focuses on effectively removing targeted data while preserving the model's general utility and knowledge (Tian et al. 2024; Wang et al. 2025b). Please refer to the appendix for more detailed related works.

Conclusion

This paper tries to bridge LLM knowledge editing and unlearning communities by studying whether editing methods are strong baselines for unlearning tasks. The findings reveal that the answer might be positive. We also explore two techniques to better adapt editing methods under unlearning setups.

Acknowledgments

This work was supported by Zhejiang Provincial Key Research and Development Project (2023C01043), Zhejiang Province Leading Geese Plan (2025C02025), Academy Of Social Governance Zhejiang University, and Inclusive and Smart Urban-Rural Governance Lab, Zhejiang University. Also supported by the following entities: The Royal Academy of Engineering via DANTE (a RAEng Chair); the European Research Council, specifically the REDIAL project; SPRIND under the composite learning challenge; and Google through a Google Academic Research Award.

References

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901.

Chang, H.; Park, J.; Ye, S.; Yang, S.; Seo, Y.; Chang, D.-S.; and Seo, M. 2024. How do large language models acquire factual knowledge during pretraining? In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Das, B. C.; Amini, M. H.; and Wu, Y. 2025. Security and privacy challenges of large language models: A survey. *ACM Computing Surveys*, 57(6): 1–39.

Fang, J.; Jiang, H.; Wang, K.; Ma, Y.; Jie, S.; Wang, X.; He, X.; and Chua, T.-S. 2025. Alphaedit: Null-space constrained knowledge editing for language models. In *The Thirteenth International Conference on Learning Representations*.

Golatkar, A.; Achille, A.; and Soatto, S. 2020. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9304–9312.

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

Guo, P.; Syed, A.; Sheshadri, A.; Ewart, A.; and Dziugaite, G. K. 2024. Mechanistic unlearning: Robust knowledge unlearning and editing via mechanistic localization. *arXiv preprint arXiv:2410.12949*.

Hartvigsen, T.; Sankaranarayanan, S.; Palangi, H.; Kim, Y.; and Ghassemi, M. 2023. Aging with grace: Lifelong model editing with discrete key-value adaptors. *Advances in Neural Information Processing Systems*, 36: 47934–47959.

Huang, J. Y.; Zhou, W.; Wang, F.; Morstatter, F.; Zhang, S.; Poon, H.; and Chen, M. 2024. Offset unlearning for large language models. *arXiv preprint arXiv:2404.11045*.

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. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Li, N.; Pan, A.; Gopal, A.; Yue, S.; Berrios, D.; Gatti, A.; Li, J. D.; Dombrowski, A.-K.; Goel, S.; Mukobi, G.; et al. 2024. The WMDP Benchmark: Measuring and Reducing Malicious Use with Unlearning. In *Forty-first International Conference on Machine Learning*.

Liu, A.; Feng, B.; Wang, B.; Wang, B.; Liu, B.; Zhao, C.; Dengr, C.; Ruan, C.; Dai, D.; Guo, D.; et al. 2024a. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. *arXiv preprint arXiv:2405.04434*.

Liu, B.; Liu, Q.; and Stone, P. 2022. Continual learning and private unlearning. In *Conference on Lifelong Learning Agents*, 243–254. PMLR.

Liu, C.; Wang, Y.; Flanigan, J.; and Liu, Y. 2024b. Large language model unlearning via embedding-corrupted prompts. *Advances in Neural Information Processing Systems*, 37: 118198–118266.

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

Maini, P.; Feng, Z.; Schwarzschild, A.; Lipton, Z. C.; and Kolter, J. Z. 2024. TOFU: A Task of Fictitious Unlearning for LLMs. In *First Conference on Language Modeling*.

Meng, K.; Bau, D.; Andonian, A.; and Belinkov, Y. 2022. Locating and editing factual associations in gpt. *Advances in neural information processing systems*, 35: 17359–17372.

Meng, K.; Sharma, A. S.; Andonian, A. J.; Belinkov, Y.; and Bau, D. 2023. Mass-Editing Memory in a Transformer. In *The Eleventh International Conference on Learning Representations*.

Mitchell, E.; Lin, C.; Bosselut, A.; Finn, C.; and Manning, C. D. 2022. Fast Model Editing at Scale. In *International Conference on Learning Representations*.

Nguyen, Q. P.; Low, B. K. H.; and Jaillet, P. 2020. Variational bayesian unlearning. *Advances in Neural Information Processing Systems*, 33: 16025–16036.

Ong, J. C. L.; Chang, S. Y.-H.; William, W.; Butte, A. J.; Shah, N. H.; Chew, L. S. T.; Liu, N.; Doshi-Velez, F.; Lu, W.; Savulescu, J.; et al. 2024. Ethical and regulatory challenges of large language models in medicine. *The Lancet Digital Health*, 6(6): e428–e432.

Pardau, S. L. 2018. The california consumer privacy act: Towards a european-style privacy regime in the united states. *J. Tech. L. & Pol'y*, 23: 68.

Qiu, X.; Shen, W. F.; Chen, Y.; Cancedda, N.; Stenetorp, P.; and Lane, N. D. 2024. Pistol: Dataset compilation pipeline for structural unlearning of llms. *arXiv preprint arXiv:2406.16810*.

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

Shen, W. F.; Qiu, X.; Kurmanji, M.; Iacob, A.; Sani, L.; Chen, Y.; Cancedda, N.; and Lane, N. D. 2025. LUNAR: LLM Unlearning via Neural Activation Redirection. *arXiv preprint arXiv:2502.07218*.

Thirunavukarasu, A. J.; Ting, D. S. J.; Elangovan, K.; Gutierrez, L.; Tan, T. F.; and Ting, D. S. W. 2023. Large language models in medicine. *Nature medicine*, 29(8): 1930–1940.

Tian, B.; Liang, X.; Cheng, S.; Liu, Q.; Wang, M.; Sui, D.; Chen, X.; Chen, H.; and Zhang, N. 2024. To Forget or Not? Towards Practical Knowledge Unlearning for Large Language Models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, 1524–1537.

Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. 2023a. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

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

Veldanda, A. K.; Zhang, S.-X.; Das, A.; Chakraborty, S.; Rawls, S.; Sahu, S.; and Naphade, M. 2024. LLM Surgery: Efficient Knowledge Unlearning and Editing in Large Language Models. *arXiv preprint arXiv:2409.13054*.

Voigt, P.; and Von dem Bussche, A. 2017. The eu general data protection regulation (gdpr). *A practical guide, 1st ed., Cham: Springer International Publishing*, 10(3152676): 10–5555.

Wang, B.; Chen, W.; Pei, H.; Xie, C.; Kang, M.; Zhang, C.; Xu, C.; Xiong, Z.; Dutta, R.; Schaeffer, R.; et al. 2023. DecodingTrust: A Comprehensive Assessment of Trustworthiness in GPT Models. In *NeurIPS*.

Wang, J.; Xu, H.; Jia, H.; Zhang, X.; Yan, M.; Shen, W.; Zhang, J.; Huang, F.; and Sang, J. 2025a. Mobile-Agent-v2: Mobile Device Operation Assistant with Effective Navigation via Multi-Agent Collaboration. *Advances in Neural Information Processing Systems*, 37: 2686–2710.

Wang, L.; Zeng, X.; Guo, J.; Wong, K.-F.; and Gottlob, G. 2025b. Selective forgetting: Advancing machine unlearning techniques and evaluation in language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 843–851.

Wang, P.; Li, Z.; Zhang, N.; Xu, Z.; Yao, Y.; Jiang, Y.; Xie, P.; Huang, F.; and Chen, H. 2024a. Wise: Rethinking the knowledge memory for lifelong model editing of large language models. *Advances in Neural Information Processing Systems*, 37: 53764–53797.

Wang, P.; Zhang, N.; Tian, B.; Xi, Z.; Yao, Y.; Xu, Z.; Wang, M.; Mao, S.; Wang, X.; Cheng, S.; et al. 2024b. EasyEdit: An Easy-to-use Knowledge Editing Framework for Large Language Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, 82–93.

Wang, Y.; Wei, J.; Liu, C. Y.; Pang, J.; Liu, Q.; Shah, A. P.; Bao, Y.; Liu, Y.; and Wei, W. 2025c. LLM Unlearning via Loss Adjustment with Only Forget Data. *ICLR*.

Yao, Y.; Wang, P.; Tian, B.; Cheng, S.; Li, Z.; Deng, S.; Chen, H.; and Zhang, N. 2023. Editing Large Language Models: Problems, Methods, and Opportunities. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 10222–10240.

Yao, Y.; Xu, X.; and Liu, Y. 2024. Large language model unlearning. *Advances in Neural Information Processing Systems*, 37: 105425–105475.

Yu, L.; Chen, Q.; Zhou, J.; and He, L. 2024. Melo: Enhancing model editing with neuron-indexed dynamic lora. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 19449–19457.

Zhang, B.; Chen, Z.; Zheng, Z.; Li, J.; and Chen, H. 2025. Resolving Editing-Unlearning Conflicts: A Knowledge Codebook Framework for Large Language Model Updating. *arXiv preprint arXiv:2502.00158*.