Dissecting Fine-Tuning Unlearning in Large Language Models

Yihuai Hong Yuelin Zou Lijie Hu Ziqian Zeng Di Wang Haiqin Yang

Abstract

Fine-tuning unlearning has been widely adopted to remove undesirable knowledge from large language models (LLMs). However, despite its practical success, the internal mechanisms underlying how fine-tuning achieves unlearning remain poorly understood. In this paper, we conduct a systematic investigation into the internal behaviors of LLMs during fine-tuning unlearning. Through activation patching and parameter restoration experiments, we analyze which layers and components are primarily responsible for the unlearning effect. Our results reveal that fine-tuning unlearning predominantly operates by altering shallow layers, while deeper layers retain much of the original knowledge. Furthermore, we identify a global negative effect induced by fine-tuning unlearning, where performance on unrelated tasks is also degraded. These findings provide new insights into the mechanisms and side effects of fine-tuning unlearning, offering guidance for developing more effective and targeted unlearning methods in future work.

1 Introduction

Large language models (LLMs), due to their extensive pre-training corpora, often inadvertently learn harmful, sensitive, or copyright-protected knowledge (Chang et al., 2023a; Mozes et al., 2023; Eldan and Russinovich, 2023; Ye et al., 2022). Consequently, recent research has focused on developing efficient unlearning methods as a post-training technique to selectively unlearn the specific knowledge (Blanco-Justicia et al., 2024; Liu et al., 2024). Currently, the core mechanism of these unlearning methods involves finetuning (Eldan and Russinovich, 2023; Jang et al., 2023; Yao et al., 2024; Rafailov et al., 2023), with corresponding adjustments and designs in the loss function to facilitate the unlearning process. Although earlier investigations (Hong et al., 2024; Lee et al., 2024a) have proven that these methods are ineffective at completely erasing model-embedded knowledge, the factors contributing to the misleading success of these techniques remain unclear.

Therefore, in this paper, we try to unveil why existing finetuning-based unlearning methods perform well in behavioral tests by analyzing the mechanisms of internal knowledge recall and flow within models (Meng et al., 2022; Pochinkov and Schoots, 2024; Geva et al., 2021a). Specifically, we investigate which components or parameters carry these unlearning effects. We design activations patching and parameters restoration experiments in three settings, aiming to independently study the impact of unlearning methods on the coefficients and value vectors in the MLPs, as well as on the attention components' states. Our findings further confirm that the methods do not truly alter the knowledge embedded in the value vectors of MLPs, and reveal that they will change how they extract and transfer this knowledge through modifications in the coefficients of MLPs and attention components during unlearning. Notably, the coefficients produced by the MLP in the final layers are primarily responsible for achieving the unlearning effects of finetuning-based methods.

We further test the global behavior impact of these fine-tuning-based unlearning methods on LLaMA2-7B-chat (Touvron et al., 2023) and OLMo-7B (Groeneveld et al., 2024) by implementing them on the respective pretraining datasets of both models, aiming to more closely simulate the erasure of knowledge acquired during the pretraining process. We discover that while these methods

appear to effectively unlearn target knowledge, they also inevitably affect the output and behavior related to unrelated knowledge. This unintended consequence stems from the fact that these approaches are based on altering the model's internal knowledge retrieval mechanisms, thereby impacting its global behavior and overall performance.

Ultimately, we conclude once again that current fine-tuning-based unlearning methods cannot completely erase sensitive knowledge embedded in models, particularly within the MLPs, instead adjusting the mechanisms by which the model retrieves knowledge. These methods are vulnerable to recovery attacks in components' activations and unsuitable for true unlearning. We advocate for future unlearning evaluations to concentrate on precise measurement of both the actual storage of targeted knowledge within the model's entire parameter set and the specific dynamics of how this knowledge is retrieved and utilized.

2 Background and Related Work

Unlearning in Large Language Models Since large language models learn knowledge from different domains and corpora during the pre-training process, it is often found that they contain harmful, sensitive or private knowledge, leading to the possibility that language models produce output behaviors containing corresponding sensitive or harmful information (Liu et al., 2024; Chang et al., 2023a; Mozes et al., 2023). Therefore, unlearning emerges as a timely and important post-pretraining processing method for LLM safety. Currently, the vast majority of LLM unlearning methods use fine-tuning as the primary operational approach. In terms of classifying them by different training objectives, they include gradient direction control (Jang et al., 2023; Yao et al., 2024, 2023) and preference optimization methods (Rafailov et al., 2023; Zhao et al., 2024; Lee et al., 2024b). In terms of classifying them by the parameters covered during training, they include full parameters fine-tuning (Eldan and Russinovich, 2023; Jang et al., 2023; Yao et al., 2024; Rafailov et al., 2023), sparse fine-tuning (Chang et al., 2023b; Stoehr et al., 2024), and parameter-efficient fine-tuning (Lu et al., 2024; Chen and Yang, 2023). Additionally, there are also a few knowledge editing methods (Patil et al., 2024). We present the specific logic details of each method in §A.

Knowledge Storation in Large Language Models Studying how knowledge is stored, transferred, and extracted in LLMs has always been an important direction in the research of LLM's interpretability (Meng et al., 2022; Geva et al., 2021b; Sukhbaatar et al., 2015; Geva et al., 2023). It is known that in transformer-based language models, the MLP is a crucial component for storing the model's factual knowledge, and its sub-layers can be viewed as key-value memories (Geva et al., 2021b). To be specific, the first layer of MLP sub-layers can be viewed as a matrix W_K formed by key vectors $k_1, k_2, \ldots, k_n$, used to capture a set of patterns in the input sequence, and ultimately outputting the coefficient scores. The second layer can be viewed as a matrix W_V formed by value vectors $v_1, v_2, \ldots, v_n$, with each value vector containing the corresponding factual knowledge (represented through token distributions). Finally, the MLP's output can be defined as the sum of value vectors weighted by their memory coefficients:

$$M_\ell = f(W_K^\ell x_\ell) W_V^\ell = m_\ell W_V^\ell, \tag{1}$$

where M_ℓ represents the output of the MLP in the transformer's ℓ-th layer for an input hidden state x_ℓ at that layer with the parameters W_K^ℓ and $W_V^\ell \in \mathbb{R}^{n \times d}$. f is a non-linearity function†. $m_\ell \in \mathbb{R}^n$ represents the coefficient scores. The dimension size of hidden states is d and it is n for the intermediate MLP.

In addition to the MLP, primarily responsible for knowledge storage, the attention component is currently considered the main component responsible for knowledge transfer and extraction in

language models (Geva et al., 2023). Here, we will not go into detail about its specific structure but only study the impact it has on knowledge extraction. The final computation formula for the hidden states in the language model is defined as:

$$X_{\ell+1} = X_\ell + M_\ell + A_\ell, \tag{2}$$

where X_ℓ, M_ℓ and A_ℓ represent the hidden states, MLP's output, and the attention component's output in the transformer's ℓ-th layer, respectively.

3 Patching Investigation

Hypothesis and Experimental Design

Based on Eq. (1) and Eq. (2), we hypothesize that there are three main reasons why the current fine-tuning-based unlearning methods appear successful in behavioral tests and seem to suggest that true unlearning has been achieved:

1. The coefficients m_ℓ are changed after fine-tuning, leading to a change in the activations of the MLPs;
2. The value vectors W_V^ℓ in MLPs are changed, causing a change in the knowledge they contain;
3. The change that happens in attention components caused the model's focus and the corresponding information extracted by these attention components A_ℓ to change, thus reducing the target knowledge-related information in the output.

Here, for the sake of simplicity and better understanding, we continue to use the definitions of m_ℓ, W_V^ℓ, and A_ℓ as given in Eq. (1) and Eq. (2) in the following. We ignore the minor effects caused by other components or parameters, such as the language model's unembedding matrix and the normalization layers. Based on the possible reasons described above, on the unlearned model, we conduct three different sets of activation patching or components' parameter restoration experiments, trying to recover the output of the target knowledge in the unlearned model. The specific operation process is as follows:

1. In the first set of experiments, we restore the coefficient scores m_ℓ corresponding to each MLP component, layer by layer, in the language model, without making any intentional changes to the value vector parameters W_V^ℓ of the MLPs or the attention components' states A_ℓ in any layer.
2. In the second set of experiments, we restore the parameters of value vectors W_V^ℓ in MLPs layer by layer, recovering the knowledge they originally contained. In this process, we avoid making intentional changes to the unlearned model's original coefficients m_ℓ and the attention components' states A_ℓ.
3. In the third set of experiments, we restore the original attention components' states A_ℓ, but without intentionally altering the MLPs' coefficient scores m_ℓ or the value vectors' parameters W_V^ℓ, only studying the impact brought by the attention components which are responsible for extracting and transferring knowledge.

To evaluate the extent of knowledge restoration, we propose the metric of Knowledge Recovery Score (KRS):

$$\text{KRS} = 1 - \frac{\text{loss}_o^*}{\text{loss}^*}, \tag{3}$$

where the losses are the average of MSE$(\cdot)$ on $L^* * i, n$ and $L * i, n$ and on $L^{*o} * i, n$ and $L * i, n$, respectively. MSE$(\cdot)$ represents the mean squared error (MSE) loss function. L, L^*, and L^{*o} are the logit distribution of the subsequent token produced by the vanilla model, unlearned model, and unlearned-then-recover model, respectively. The average loss is computed on the next I generated tokens on N knowledge-related questions.

Finally, if KRS approaches 1, it indicates $L^{*o} * i, n$ and $L * i, n$ that are nearly consistent, representing a higher degree of knowledge recovery. Conversely, a lower KRS suggests a lower degree of that.

Activation Patching and Parameters Restoration Experiments

We conduct the experiments on two recent LLMs, LLaMA2-7B-chat (Touvron et al., 2023) and OLMo-7B (Groeneveld et al., 2024). We apply two example finetuning-based unlearning methods, DPO (Rafailov et al., 2023) and Gradient Difference (Yao et al., 2024), to perform unlearning on the large language models and calculate the average KRS scores. Inspired by (Eldan and Russinovich, 2023), which tries to unlearn the concept knowledge of "Harry Potter" in language models, we extend this experiment by selecting 10 well-known concepts per model from the ConceptVectors Benchmark (Hong et al., 2024), which is a collection of concepts that language models are well-acquainted with and have substantial knowledge about. Examples of them are provided in Table 2 of §B. For the unlearning training, we use the texts containing the corresponding concepts from Redpjama‡ and Dolma (Soldaini et al., 2024). Redpjama is a replication of the pretraining corpus for the LLaMA model, while Dolma is the open-source pre-training dataset for the OLMo model. Detailed information is provided in §B. So here we can ensure that the knowledge to be unlearned was at least seen by the model during the pre-training process, and that the training data used more broadly covers the textual sources from which the model acquired the corresponding knowledge about certain concepts.

After obtaining the unlearned model, we follow the steps mentioned in the hypothesis to perform activation patching and parameter restoration experiments on the unlearned models. To calculate the Knowledge Recover Scores, we set I to 30 and N to 10, indicating the generation of the next 30 tokens and the selection of 10 questions related to each concept. To make the recovery effects more pronounced and the whole process easier to observe, we adopt techniques from (Meng et al., 2022, 2023) which implemented causal mediation, setting the size of the recovery window to five. This allows us to observe the average effects of recovering five consecutive layers at a time. Details can be found in §B.

The specific results are shown in Fig. 1. From our analysis, surprisingly, we observe that when we solely recover the parameters contained in the value vectors of each layer in the unlearned model without interfering with the coefficients or attention components' states, the recovery of the target knowledge is negligible (The KRS scores are all below 0.001). This holds regardless of which layer is recovered, and regardless of the specific model being considered.

However, when recovering the attention components' states in the intermediate layers (from the 15th layer onward) or deeper layers (from the 27th layer onward), we can observe that the average KRS for both models has increased to exceed 0.3 and 0.4, respectively, indicating that a significant portion of the corresponding knowledge has been recovered. What's more, restoring the coefficients

of the MLPs in the intermediate layers (from the 20th layer onward) and deeper layers (from the 29th layer) also yields impressive knowledge recovery effects.

The layers at which the scores start to increase under the two settings generally align closely with the observation by Geva et al. (2023) that the MLP modules recall knowledge in intermediate layers, and the attention components mostly start to extract and transfer information in the deeper layers, or after the model has completed the relevant knowledge recall. We also tried simultaneously recovering the coefficients and attention states and found that the model can achieve much greater knowledge recovery, with the peak KRS score exceeding 0.9 on both models.

Additionally, it is noteworthy that, simply restoring the coefficient scores of the MLP outputs from the last two or three layers can significantly elevate the KRS of the unlearned LLaMA and OLMo models to 0.8 or above. This suggests that the coefficient scores of the MLPs in the last layers might play a crucial role in the final behavior results of the LLM. To better isolate the effects of restoring m_ℓ, W_V^ℓ, and A_ℓ individually and support the above argument, we present a more rigorous patching and restoration experiment in §C, with the corresponding results shown in Figure 3. Ultimately, we found that the restoration of the attention states also contributed to the coefficients of the MLP in the final layers, further confirming that these coefficients carry the primary role of achieving the effects of finetuning-based unlearning. It also indicates that fine-tuning largely adjusts the model's behavior by modifying the coefficients of the deep MLP layers, likely because this enables faster adaptation compared to other knowledge adjustment mechanisms, such as altering knowledge encoded in the MLP itself. This phenomenon and the potential defensive strategy have not been discussed in the previous literature, warranting further investigation in future studies.

Overall, these results all further confirm that the finetuning-based unlearning methods essentially do not modify the model knowledge contained in the value vectors, but adjust the way knowledge is called during the fine-tuning process, either by adjusting the coefficients to modulate the MLP activation or by adjusting the attention to extract and transfer knowledge.

4 Global Negative Effect of Fine-Tuning Unlearning

In the previous section, we demonstrated that these finetuning-based methods alter the model's final behavior by adjusting the MLP output coefficients in the final layers. Therefore, we hypothesize that this behavioral change will has a global effect, potentially impacting the output of unrelated knowledge as well. In this section, we verify this hypothesis through the following experiments.

We apply four fine-tuning-based unlearning methods to the concepts used in §3 on their pretraining text sources (from RedPajama and Dolma) with the goal of erasing the learned knowledge during pretraining through a reverse process. These methods are as follows: DPO (Rafailov et al., 2023), NPO (Zhao et al., 2024), NPO+KL (Zhao et al., 2024) and Gradient Difference (Yao et al., 2024). The details of these baselines and data statistics are shown in §A and §B. We evaluate the unlearning effectiveness of these methods on the concepts' related QA pairs and the unlearning impact on unrelated QA pairs, reporting the average scores of BLEU (Papineni et al., 2002) by comparing the model's response before and after unlearning. In Figure 2, we report their performance at the end of each training epoch respectively.

We can observe that for finetuning-based methods, as the number of training epochs increases, aiming to achieve a lower target QA BLEU score, the corresponding unrelated QA BLEU score also decreases accordingly, exhibiting a positive correlation. This suggests that the impact of finetuning-based methods on the model's output behavior is global. While unlearning the target knowledge, they inadvertently alter the output behavior or manner for unrelated knowledge to a

certain degree.

5 Discussion and Conclusion

We have deeply investigated the reasons why fine-tuning-based unlearning methods seemingly succeeded in behavior-based testing for large language model unlearning: Through activation patching and parameter restoration experiments, we find that these methods alter the way knowledge is extracted by changing MLP activations or model's attention, ultimately affecting the output. This is evidenced by the fact that the model's output regarding the target knowledge is largely restored after patching the activations and the attention components' states. Furthermore, we conduct experiments on the pretraining datasets of two models, to test the models' capabilities after unlearning, verifying that in addition to unlearning the corresponding knowledge, fine-tuning-based methods that by altering the way the model accesses knowledge, will significantly impair the model's other unrelated capabilities, causing a certain degree of capability degradation.

6 Limitations

In the experiments detailed in §3, we have disregarded the potential unlearning impact caused by parameter changes in other model components during the fine-tuning process. This decision is based on the observation that the impact of such changes appears to be minimal. For instance, during our parameter comparison analysis, we found that the changes in the unembedding matrix and normalization layer parameters resulted in cosine similarity values above 0.999. This suggests that the modifications to these components are quite small in magnitude.

However, it remains unclear whether even such minimal parameter changes can still have any meaningful effect on the model's overall behavior and knowledge. Further verification and analysis would be needed to conclusively determine the extent to which these ancillary parameter updates might influence the unlearning outcome.

Acknowledgements

The work was fully supported by the IDEA Information and Super Computing Centre (ISCC), National Natural Science Foundation of China (Grant No. 62406114), the Guangzhou Basic and Applied Basic Research Foundation (Grant No. 2023A04J1687), and the Fundamental Research Funds for the Central Universities (Grant No. 2024ZYGXZR074). Di Wang and Lijie Hu are supported in part by the funding BAS/1/1689-01-01, URF/1/4663-01-01, REI/1/5232-01-01, REI/1/5332-01-01, and URF/1/5508-01-01 from KAUST, and funding from KAUST - Center of Excellence for Generative AI, under award number 5940.

References

Alberto Blanco-Justicia, Najeeb Jebreel, Benet Manzanares, David Sánchez, Josep Domingo-Ferrer, Guillem Collell, and Kuan Eeik Tan. 2024. Digital forgetting in large language models: A survey of unlearning methods. *Preprint*, arXiv:2404.02062.

Kent K. Chang, Mackenzie Hanh Cramer, Sandeep Soni, and David Bamman. 2023a. Speak, memory: An archaeology of books known to chatGPT/GPT-4. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Ting-Yun Chang, Jesse Thomason, and Robin Jia. 2023b. Do localization methods actually localize memorized data in llms? *arXiv preprint* arXiv:2311.09060.

Jiaao Chen and Diyi Yang. 2023. Unlearn what you want to forget: Efficient unlearning for llms. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12041–12052.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. 2021. Evaluating large language models trained on code. *arXiv preprint* arXiv:2107.03374.

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

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. 2023. Dissecting recall of factual associations in auto-regressive language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12216–12235.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021a. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5484–5495, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021b. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5484–5495.

Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, Shane Arora, David Atkinson, Russell Authur, Khyathi Raghavi Chandu, Arman Cohan, Jennifer Dumas, Yanai Elazar, Yuling Gu, Jack Hessel, Tushar Khot, William Merrill, Jacob Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew E. Peters, Valentina Pyatkin, Abhilasha Ravichander, Dustin Schwenk, Saurabh Shah, Will Smith, Emma Strubell, Nishant Subramani, Mitchell Wortsman, Pradeep Dasigi, Nathan Lambert, Kyle Richardson, Luke Zettlemoyer, Jesse Dodge, Kyle Lo, Luca Soldaini, Noah A. Smith, and Hannaneh Hajishirzi. 2024. Olmo: Accelerating the science of language models. *arXiv preprint* arXiv:2402.00838.

Yihuai Hong, Lei Yu, Haiqin Yang, Shauli Ravfogel, and Mor Geva. 2024. Intrinsic evaluation of unlearning using parametric knowledge traces. *Preprint*, arXiv:2406.11614.

Joel Jang, Dongkeun Yoon, Sohee Yang, Sungmin Cha, Moontae Lee, Lajanugen Logeswaran, and Minjoon Seo. 2023. Knowledge unlearning for mitigating privacy risks in language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14389–14408, Toronto, Canada. Association for Computational Linguistics.

Andrew Lee, Xiaoyan Bai, Itamar Pres, Martin Wattenberg, Jonathan K. Kummerfeld, and Rada Mihalcea. 2024a. A mechanistic understanding of alignment algorithms: A case study on DPO and toxicity. In *Forty-first International Conference on Machine Learning*.

Andrew Lee, Xiaoyan Bai, Itamar Pres, Martin Wattenberg, Jonathan K Kummerfeld, and Rada Mihalcea. 2024b. A mechanistic understanding of alignment algorithms: A case study on dpo and toxicity. *arXiv preprint* arXiv:2401.01967.

Sijia Liu, Yuanshun Yao, Jinghan Jia, Stephen Casper, Nathalie Baracaldo, Peter Hase, Xiaojun Xu, Yuguang Yao, Hang Li, Kush R. Varshney, Mohit Bansal, Sanmi Koyejo, and Yang Liu. 2024. Rethinking machine unlearning for large language models. *Preprint*, arXiv:2402.08787.

Weikai Lu, Ziqian Zeng, Jianwei Wang, Zhengdong Lu, Zelin Chen, Huiping Zhuang, and Cen Chen. 2024. Eraser: Jailbreaking defense in large language models via unlearning harmful knowledge. *arXiv preprint* arXiv:2404.05880.

Kevin Meng, David Bau, Alex J Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*.

Kevin Meng, Arnab Sen Sharma, Alex J Andonian, Yonatan Belinkov, and David Bau. 2023. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations*.

Maximilian Mozes, Xuanli He, Bennett Kleinberg, and Lewis D. Griffin. 2023. Use of llms for illicit purposes: Threats, prevention measures, and vulnerabilities. *Preprint*, arXiv:2308.12833.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Vaidehi Patil, Peter Hase, and Mohit Bansal. 2024. Can sensitive information be deleted from LLMs? objectives for defending against extraction attacks. In *The Twelfth International Conference on Learning Representations*.

Nicholas Pochinkov and Nandi Schoots. 2024. Dissecting language models: Machine unlearning via selective pruning. *Preprint*, arXiv:2403.01267.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. OpenAI blog.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Authur, Ben Bogin, Khyathi Chandu, Jennifer Dumas, Yanai Elazar, et al. 2024. Dolma: An open corpus of three trillion tokens for language model pretraining research. *arXiv preprint* arXiv:2402.00159.

Niklas Stoehr, Mitchell Gordon, Chiyuan Zhang, and Owen Lewis. 2024. Localizing paragraph memorization in language models. *Preprint*, arXiv:2403.19851.

Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. 2015. End-to-end memory networks. *Advances in neural information processing systems*, 28.

Hugo Touvron, Louis Martin, Kevin R. Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, D. Bikel, Lukas Blecher, Cristian Cantón Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, A. Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel M. Kloumann, A. Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, R. Subramanian, Xia Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zhengxu Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint* arXiv:2307.09288.

Yuanshun Yao, Xiaojun Xu, and Yang Liu. 2023. Large language model unlearning. In *Socially Responsible Language Modelling Research*.

Yuanshun Yao, Xiaojun Xu, and Yang Liu. 2024. Large language model unlearning. *Preprint*, arXiv:2310.10683.

Jingwen Ye, Yifang Fu, Jie Song, Xingyi Yang, Songhua Liu, Xin Jin, Mingli Song, and Xinchao Wang. 2022. Learning with recoverable forgetting. In *European Conference on Computer Vision*, pages 87–103. Springer.

Weixiang Zhao, Yulin Hu, Zhuojun Li, Yang Deng, Yanyan Zhao, Bing Qin, and Tat-Seng Chua. 2024. Towards comprehensive and efficient post safety alignment of large language models via safety

patching. *arXiv preprint* arXiv:2405.13820.

A Details in Existing Unlearning Methods

In this section, we provide a more detailed introduction to the LLM unlearning methods we used in §3 and §4.

- **Gradient Difference** (Yao et al., 2024), based on Gradient Ascent, it adds a regularization term to minimize the KL divergence between the unlearned and the original LLM on a reference text dataset, thus preventing the model from catastrophic deterioration of its general capability.
- **Direct Preference Optimization (DPO)** (Rafailov et al., 2023), which maximizes the log-likelihood ratio between generating the preferred and the unfavored responses, while retaining a small shift from the original LLM predictive distribution.
- **Negative Preference Optimization (NPO)** (Zhao et al., 2024), which discards the favored responses and only minimizes the prediction probability of the unfavored answers.
- **NPO+KL** which adds to NPO a KL divergence loss between the model's outputs before and after unlearning.

B Unlearning Experiment's Corpus

Here, we present detailed information about the data used for activation patching experiments and the unlearning experiments conducted in §3 and §4. We select 10 well-known concepts from ConceptVectors Benchmark (Hong et al., 2024) and extract 6,000 corresponding training data segments containing knowledge about the respective concepts per model from the pre-training datasets of Redpjama and Dolma. These extracted data segments are used for unlearn training of the two models respectively. For each concept, we also include ten related questions from the ConceptVectors Benchmark, along with 50 unrelated questions sampled from other unrelated concepts. These are used in §4 to evaluate the unlearning effectiveness from the behavior perspective on the specific concepts, as well as to assess whether the model's unrelated capabilities were affected. We have manually checked and verified that the vanilla LLaMA and OLMo models can accurately answer these selected questions, indicating that the models possess the knowledge. All the statistics and examples are shown in Table 1 and Table 2, respectively.

C More Rigorous Patching Investigation

In §3, during our activation patching and parameters restoration experiments, we restore m_ℓ, W_V^ℓ, or A_ℓ layer by layer respectively, while avoiding intentional changes to the other two states in the unlearned model. However, for instance, restoring A_ℓ in ℓ-th layer may aid in the recovery of m_ℓ in subsequent layers, ultimately leading to an improvement in KRS. Therefore, in this part of the experiment, when restoring each element layer by layer, we purposefully keep the other two elements unchanged (e.g., when restoring A_ℓ, we maintain the original states of m_ℓ and W_V^ℓ for both the current and subsequent layers). This approach thoroughly isolates the effects of these three different elements.

Figure 3 presents the results in this setting. We can observe the following: (1) When W_V^ℓ is restored layer by layer, its effect on improving KRS remains very small, which is consistent with prior experiments. (2) When restoring A_ℓ layer by layer and isolating its effects from the other two factors, its contribution to KRS remains insignificant, staying at a low level and only increasing to around 0.08 on LLaMA and 0.2 on OLMO in the final layers. (3) When m_ℓ is restored layer by layer, isolating its influence from the other elements, we observe a notable rise in KRS in the last three layers, reaching values as high as 0.8 or above. This supports the idea that neurons responsible for m_ℓ in the MLP components of the final layers primarily carry the unlearning effects of these finetuning-based methods.