Mr. Snuffleupagus at SemEval-2025 Task 4: Unlearning Factual Knowledge from LLMs Using Adaptive RMU

Arjun Dosajh[*]
IIIT Hyderabad
arjun.dosajh@research.iiit.ac.in

Mihika Sanghi[*]
IIIT Hyderabad
mihika.sanghi@research.iiit.ac.in

Abstract

Large Language Models (LLMs) have demonstrated remarkable capabilities in natural language understanding and generation. However, their tendency to memorize training data raises concerns regarding privacy, copyright compliance, and security, particularly in cases involving Personally Identifiable Information (PII). Effective machine unlearning techniques are essential to mitigate these risks, yet existing methods remain underdeveloped for LLMs due to their open-ended output space. In this work, we apply the Adaptive Representation Misdirection Unlearning (RMU) technique to unlearn sensitive information from LLMs. Through extensive experiments, we analyze the effects of unlearning across different decoder layers to determine the most effective regions for sensitive information removal. Our technique ranked 4th on the official leaderboard of both 1B parameter and 7B parameter models.

1 Introduction

In the realm of large language models (LLMs), unlearning is particularly challenging due to the highly distributed nature of knowledge storage across model parameters. Unlike Computer Vision or Graph Neural Networks (Chundawat et al., 2023; Kolipaka et al., 2024), where feature representations tend to be more localized, LLMs encode knowledge in an interwoven manner, making targeted removal complex (Meng et al., 2022). The need for effective LLM unlearning arises from concerns surrounding data privacy, bias mitigation, and compliance with regulations such as GDPR's 'right to be forgotten'. If an LLM generates sensitive or misleading information, it is crucial to develop mechanisms that remove such knowledge without degrading overall performance (Eldan and Russinovich, 2023).

An approach to tackling unlearning in LLMs involves model-editing, a technique closely tied to mechanistic interpretability. Model-editing methods aim to modify the internal representations of an LLM to suppress or alter specific outputs while maintaining overall fluency and coherence. Previous work has explored various approaches, such as neuron pruning (Frankle and Carbin, 2019) and gradient-based forgetting (Yao et al., 2023; Goodfellow et al., 2015), to achieve unlearning without full retraining. Unlearning techniques for LLMs often lead to catastrophic forgetting, compromising general performance (Luo et al., 2025; Zhang et al., 2024; Kemker et al., 2018; ROBINS, 1995). Understanding how and where knowledge is stored in an LLM is crucial for designing effective unlearning strategies. Research has shown that different decoder layers in transformer-based architectures capture different types of information.

The task is outlined in the task description paper (Ramakrishna et al., 2025b). There are separate leaderboards for the 1B and 7B parameter models.

Participating in this task provided valuable insights into both the strengths and limitations of our system. Quantitatively, our approach achieved competitive results, ranking 4th among participating teams on a metric designed specifically for this task. Additionally, we analyze how factual information is distributed across different decoder layers in large language models (LLMs). Through our experiments, we demonstrate that unlearning factual information from middle-later layers is particularly effective.

We have released the code for our system on GitHub[1], facilitating transparency and reproducibility in our approach.

2 Related works

Effective unlearning techniques focus on modifying model representations to diminish the influence

[*] Equal contribution.

[1] https://github.com/ArjunDosajh/Mr.Snuffleupagus-SemEval-2025-Task-4

Proceedings of the The 19th International Workshop on Semantic Evaluation (SemEval-2025), pages 1616–1622
July 31 - August 1, 2025 ©2025 Association for Computational Linguistics

of specific data while preserving overall performance. These methods often involve fine-tuning strategies that steer internal representations away from unwanted content, ensuring that the model forgets particular information without compromising its general capabilities.

One prominent approach is Representation Misdirection Unlearning (RMU) (Li et al., 2024), which directs the model's intermediate representations of data intended for unlearning toward a predetermined random vector. This technique effectively reduces the model's performance on tasks related to the forgotten content while maintaining its proficiency in other domains. RMU has been demonstrated to lower the model's knowledge of the Weapons of Mass Destruction Proxy (WMDP) dataset, indicating its potential in reducing malicious use of LLMs (Li et al., 2024).

Building upon RMU, Adaptive RMU introduces a dynamic adjustment mechanism for the steering coefficient, which influences the alignment of forget-sample representations with the random direction (Huu-Tien et al., 2025). This adaptive approach enhances unlearning effectiveness across various network layers, addressing limitations observed when RMU is applied to middle and later layers of LLMs. Adaptive RMU not only improves unlearning performance but also maintains robustness against adversarial attacks, ensuring that the model does not inadvertently relearn the forgotten content.

Another innovative technique involves the use of Sparse Autoencoders (SAEs) to remove specific knowledge from LLMs (Farrell et al., 2024). By training SAEs to capture and subsequently eliminate features associated with the content to be unlearned, this method effectively reduces the model's ability to recall unwanted information. Studies have shown that applying SAEs can unlearn subsets of sensitive data with minimal impact on the model's performance in other areas, offering a targeted and efficient unlearning strategy.

3 Background

3.1 Task

The challenge spans three subtasks evaluating unlearning: long-form synthetic creative texts across genres; short-form synthetic biographies containing PII (such as fake names, phone numbers, SSNs, email addresses, and home addresses); and real documents drawn from the model's training data. Each subtask involves both sentence-completion and question-answering evaluations. In both cases, the dataset is divided into a retain set (which should be preserved) and a forget set (which should be unlearned). The goal is for the model to behave as if trained solely on the retain set, excluding the forget set, thereby mimicking an ideal unlearning scenario (Ramakrishna et al., 2025b).

3.2 Dataset

The dataset quantifies unlearning performance for each subtask and evaluation type, with separate retain and forget sets whose sizes are summarized in Table 1 (Ramakrishna et al., 2025a).

Subtask	Forget Set Size	Retain Set Size
1	214	260
2	780	762
3	372	392

Table 1: Dataset sizes for retain and forget sets across all subtasks.

3.3 Objective

The challenge aims to develop and evaluate effective unlearning techniques for large language models while preserving overall model performance. Submissions are assessed based on their ability to forget specified information while retaining non-targeted knowledge.

Performance is measured through three metrics:

- **Task-Specific Regurgitation Rates:** This combines ROUGE-L for sentence completion and exact-match for question answering, evaluated on both the retain and forget sets. Forget-set scores are inverted as $1 -$ value, and the 12 resulting scores are aggregated via the harmonic mean.

- **Membership Inference Attack (MIA) Score:** Calculated as

$$1 - \left|\text{mia\_loss\_auc\_score} - 0.5\right| \times 2,$$

this metric assesses privacy leakage following (Shokri et al., 2017).

- **MMLU Benchmark Performance:** Accuracy on the 57-subject multiple-choice MMLU suite measures language understanding and reasoning (Hendrycks et al., 2021).

The final score is the arithmetic mean of these three metrics.

3.4 Model

For our experiments, we use fine-tuned versions of the `OLMo-7B-0724-Instruct-hf` and `OLMo-1B-0724-hf` models. OLMo (Open Language Model) is an open-source language model developed to facilitate research in language modeling and knowledge retention (Groeneveld et al., 2024). Both models follow a decoder-only transformer architecture, similar to traditional autoregressive language models. The task organizers provided the fine-tuned models.

The `OLMo-7B` model consists of **32 decoder layers** with a hidden size of 4096 and 32 attention heads. The smaller `OLMo-1B` model has **16 decoder layers**, a hidden size of 2048, and 16 attention heads. These models have been specifically fine-tuned to memorize documents from all three tasks, making them a suitable testbed for evaluating unlearning methods.

4 System Overview

4.1 RMU

Representation Misdirection Unlearning (RMU) is a technique designed to selectively unlearn hazardous knowledge from a large language model (LLM) while preserving its general capabilities. It employs a two-part loss function: a *forget loss* that degrades harmful representations and a *retain loss* that ensures minimal disruption to benign knowledge.

The forget loss modifies model activations at a specific layer ℓ by increasing their norm in a fixed random direction. Given:

- $M_u(\cdot)$: hidden states of the unlearned model
- $M_f(\cdot)$: hidden states of the original frozen model
- $\mathbf{u}$: a fixed random unit vector sampled from $[0, 1]$
- $\mathcal{D}_f$: forget dataset

The forget loss is defined as:

$$\mathcal{L}_{\text{forget}} = \mathbb{E}_{x_f \sim \mathcal{D}_f} \left[\frac{1}{L_f} \sum_{t \in x_f} \| M_u(t) - c \cdot \mathbf{u} \|^2 \right], \tag{1}$$

where L_f is the number of tokens in x_f, and c is a scaling hyperparameter.

To prevent excessive forgetting, RMU introduces a retain loss that regularizes the unlearned model activations to remain close to those of the frozen model on the retain dataset $\mathcal{D}_r$:

$$\mathcal{L}_{\text{retain}} = \mathbb{E}_{x_r \sim \mathcal{D}_r} \left[\frac{1}{L_r} \sum_{t \in x_r} \| M_u(t) - M_f(t) \|^2 \right], \tag{2}$$

where L_r is the number of tokens in x_r.

The final loss function is a weighted combination of the forget and retain losses:

$$\mathcal{L} = \mathcal{L}_{\text{forget}} + \alpha \cdot \mathcal{L}_{\text{retain}}, \tag{3}$$

Where α controls the trade-off between forgetting and retention. RMU updates model weights iteratively, focusing on layers $\ell - 2, \ell - 1$, and ℓ to improve efficiency.

In the original RMU implementation, the authors use an external dataset, such as WikiText, to preserve the model's general capabilities. Instead, we adapt the retain dataset to match our task-specific retain set and demonstrate that RMU remains effective.

4.2 Adaptive RMU

Building on RMU, Adaptive RMU introduces a modified forget loss by scaling the random unit vector $\mathbf{u}$ with an *adaptive scaling coefficient* $\beta \| h^{(l)}_{\theta_{\text{frozen}}}(x_F) \|$. Here, $\beta \in \mathbb{R}$ is a scaling factor, and $\| h^{(l)}_{\theta_{\text{frozen}}}(x_F) \|$ is the ℓ_2-norm of the forget sample x_F in the frozen model $f_{\theta_{\text{frozen}}}$. This ensures that the magnitude of the perturbation adapts to the norm of the activation, leading to a more stable unlearning process. The total loss in Adaptive RMU is given by:

$$\begin{aligned} \mathcal{L}^{\text{adaptive}} = \mathbb{E}_{x_f \sim \mathcal{D}_\mathrm{f}} \left[\frac{1}{L_f} \sum_{t \in x_f} \left\| M_\mathrm{u}(t) - \beta \| M_\mathrm{f}(t) \| \mathbf{u} \right\|_2^2 \right] \\ + \alpha \mathbb{E}_{x_r \sim \mathcal{D}_\mathrm{r}} \left[\frac{1}{L_r} \sum_{t \in x_r} \left\| M_\mathrm{u}(t) - M_\mathrm{f}(t) \right\|_2^2 \right] \end{aligned} \tag{4}$$

Figure 1 illustrates the construction of the loss function with adaptive scaling. This adaptation makes the forgetting process proportional to the activation strength of the original model while ensuring that general capabilities are preserved through the retain loss.

To account for the varying dataset sizes across different splits (creative documents, sensitive content,

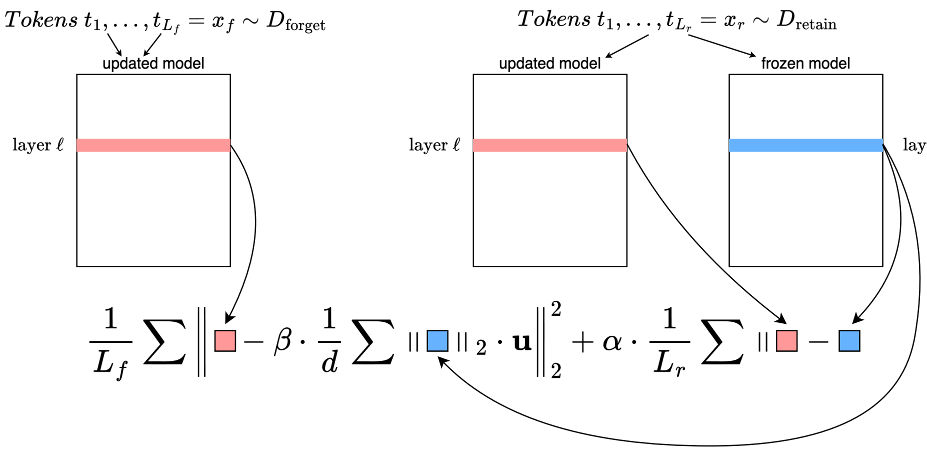

Tokens $t_1, \ldots, t_{L_f} = x_f \sim D_{\text{forget}}$
updated model
layer ℓ
Tokens $t_1, \ldots, t_{L_r} = x_r \sim D_{\text{retain}}$
updated model
frozen model
layer ℓ
layer ℓ
$\frac{1}{L_f} \sum \left\| 🟥 - \beta \cdot \frac{1}{d} \sum \| 🟦 \|_2 \cdot \mathbf{u} \right\|_2^2 + \alpha \cdot \frac{1}{L_r} \sum \| 🟥 - 🟦$

Figure 1: RMU performs machine unlearning by optimizing a two-part loss: a forget term, which misdirects the model's activations for forget-set inputs, and a retain term, which aligns the model's activations for retain-set inputs. The random vector $\boldsymbol{u}$ is adaptively scaled using the frozen model's activations, $\boldsymbol{d}$ denotes the embedding dimension of the model (Li et al., 2024; Huu-Tien et al., 2025)

and real documents), we use a randomized sampling approach instead of alternating uniformly between them. In each training step, a sample from both the forget and retain datasets is selected, tokenized, and processed by the model. To guide unlearning, each forget sample is assigned a control vector, which is scaled using a predefined steering coefficient. The unlearning loss is computed as the MSE between the model's activations and the control vector, with an adaptive coefficient that dynamically scales based on the activation norms.

Due to computational constraints, our experiments were conducted exclusively on the 1B parameter model. However, our approach demonstrates competitive performance on the 7B parameter model as well.

5 Experiments

5.1 Preprocessing

Our preprocessing pipeline involves tokenizing the text using the allenai/OLMo-1B-0724-hf tokenizer, which has a vocabulary size of 50,280. This tokenizer includes specialized tokens for personally identifiable information (PII), such as email addresses and Social Security Numbers (SSNs), ensuring a structured representation of such entities within the model.

5.2 Hyperparameter Tuning

Since the adaptive RMU approach dynamically adjusts the scaling coefficient, we primarily focus on selecting the layers for unlearning. Specifically, we experiment with all possible combinations of three consecutive layers, ranging from (0,1,2) to (13,14,15). This allows us to identify the most effective layer range for minimizing interference with retained knowledge while ensuring effective unlearning.

6 Results

We evaluated the performance of our unlearned model using the evaluation metric described in Section 3.3. Table 3 compares the performance of our approach with some baseline methods. The results of our experiments, including the task aggregate score, membership inference attack (MIA) score, and MMLU score across different layer combinations, are presented in Table 2. We find that ideal layers for unlearning with adaptive RMU are 12,13,14 for the 1B parameter model and 24,25,26 for the 7B parameter model. We conducted our experiments using four NVIDIA RTX 3090 GPUs, each equipped with 24GB of VRAM. Our approach ranked 4th among all competing teams on both the 1B parameter model leaderboard and the 7B parameter model leaderboard.

7 Conclusion

In this work, we demonstrate that applying the adaptive RMU (Rank-One Model Update) technique to later decoder layers is an effective strategy for unlearning factual information from large language models (LLMs). A key advantage of our method is its minimal hyperparameter tuning requirement, making it easily adaptable to different LLM architectures.

Decoder Layers	Task Aggregate	MIA	MMLU	Final score
0,1,2	0.547	0.062	0.244	0.284
1,2,3	0.542	0.081	0.249	0.291
2,3,4	0.355	0.401	0.250	0.336
3,4,5	0.433	0.490	0.254	0.392
4,5,6	0.508	0.355	0.229	0.364
5,6,7	**0.637**	0.357	0.262	0.419
6,7,8	0.597	0.416	0.250	0.421
7,8,9	0.616	0.332	0.245	0.398
8,9,10	0.631	0.362	**0.265**	0.419
9,10,11	0.574	0.471	0.264	0.437
10,11,12	0.282	0.279	0.243	0.268
11,12,13	0.582	0.489	0.254	0.442
12,13,14	0.565	**0.835**	0.261	**0.554**
13,14,15	0.538	0.747	0.258	0.515

Table 2: Performance of different layer combinations on task aggregate, MIA score, and MMLU score for OLMo-1B.

Method	Task Aggregate	MIA	MMLU	Final score
~~Gradient Ascent~~	~~0~~	~~0.912~~	~~0.269~~	~~0.394~~
~~Gradient Difference~~	~~0~~	~~0.382~~	~~0.348~~	~~0.243~~
~~KL Minimization~~	~~0~~	~~0.916~~	~~0.269~~	~~0.395~~
Negative Preference Optimization	0.021	0.080	0.463	0.188
Adaptive RMU	0.387	0.872	0.485	0.376

Table 3: Comparison with baseline methods for OLMo-7B. Some methods have been striked out because MMLU score is below the minimum threshold of 0.371.

Our technique achieved 4th place on the official leaderboard for both 1B and 7B parameter models, highlighting its competitiveness among various unlearning approaches. While the original RMU implementation demonstrated that unlearning from earlier layers effectively removes hazardous knowledge—such as information related to cybersecurity threats and bioweapons, which require deep contextual understanding—we extend this research by showing that factual knowledge (e.g., phone numbers, Social Security Numbers, and addresses) is more effectively unlearned from later decoder layers.

Despite these promising results, unlearning sensitive content from LLMs remains an open challenge, with several promising directions for future research. For instance, further exploration is needed to assess the trade-offs between unlearning effectiveness and generalization, particularly when removing factual knowledge versus conceptual reasoning.

Overall, our work contributes to the broader fields of model editing and mechanistic interpretability, providing valuable insights into the layerwise dynamics of unlearning and paving the way for more robust and efficient strategies in the future.

References

Vikram S. Chundawat, Ayush K. Tarun, Murari Mandal, and Mohan Kankanhalli. 2023. Zero-shot machine unlearning. IEEE Transactions on Information Forensics and Security, 18:2345–2354.

Ronen Eldan and Mark Russinovich. 2023. Who's harry potter? approximate unlearning in llms.

Eoin Farrell, Yeu-Tong Lau, and Arthur Conmy. 2024. Applying sparse autoencoders to unlearn knowledge in language models.

Jonathan Frankle and Michael Carbin. 2019. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In International Conference on Learning Representations.

Ian J. Goodfellow, Mehdi Mirza, Da Xiao, Aaron Courville, and Yoshua Bengio. 2015. An empirical investigation of catastrophic forgetting in gradient-based neural networks.

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.

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

Dang Huu-Tien, Trung-Tin Pham, Hoang Thanh-Tung, and Naoya Inoue. 2025. On effects of steering latent representation for large language model unlearning.

Ronald Kemker, Marc McClure, Angelina Abitino, Tyler Hayes, and Christopher Kanan. 2018. Measuring catastrophic forgetting in neural networks. Proceedings of the AAAI Conference on Artificial Intelligence, 32(1).

Varshita Kolipaka, Akshit Sinha, Debangan Mishra, Sumit Kumar, Arvindh Arun, Shashwat Goel, and Ponnurangam Kumaraguru. 2024. A cognac shot to forget bad memories: Corrective unlearning in gnns.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D. Li, Ann-Kathrin Dombrowski, Shashwat Goel, Long Phan, Gabriel Mukobi, Nathan Helm-Burger, Rassin Lababidi, Lennart Justen, Andrew B. Liu, Michael Chen, Isabelle Barrass, Oliver Zhang, Xiaoyuan Zhu, Rishub Tamirisa, Bhrugu Bharathi, Adam Khoja, Zhenqi Zhao, Ariel Herbert-Voss, Cort B. Breuer, Samuel Marks, Oam Patel, Andy Zou, Mantas Mazeika, Zifan Wang, Palash Oswal, Weiran Lin, Adam A. Hunt, Justin Tienken-Harder, Kevin Y. Shih, Kemper Talley, John Guan, Russell Kaplan, Ian Steneker, David Campbell, Brad Jokubaitis, Alex Levinson, Jean Wang, William Qian, Kallol Krishna Karmakar, Steven Basart, Stephen Fitz, Mindy Levine, Ponnurangam Kumaraguru, Uday Tupakula, Vijay Varadharajan, Ruoyu Wang, Yan Shoshitaishvili, Jimmy Ba, Kevin M. Esvelt, Alexandr Wang, and Dan Hendrycks. 2024. The wmdp benchmark: Measuring and reducing malicious use with unlearning.

Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. 2025. An empirical study of catastrophic forgetting in large language models during continual fine-tuning.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in gpt. In Proceedings of the 36th International Conference on Neural Information Processing Systems, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Anil Ramakrishna, Yixin Wan, Xiaomeng Jin, Kai-Wei Chang, Zhiqi Bu, Bhanukiran Vinzamuri, Volkan Cevher, Mingyi Hong, and Rahul Gupta. 2025a. Lume: Llm unlearning with multitask evaluations. arXiv preprint arXiv:2502.15097.

Anil Ramakrishna, Yixin Wan, Xiaomeng Jin, Kai-Wei Chang, Zhiqi Bu, Bhanukiran Vinzamuri, Volkan Cevher, Mingyi Hong, and Rahul Gupta. 2025b. Semeval-2025 task 4: Unlearning sensitive content from large language models. arXiv preprint.

ANTHONY ROBINS. 1995. Catastrophic forgetting, rehearsal and pseudorehearsal. Connection Science, 7(2):123–146.

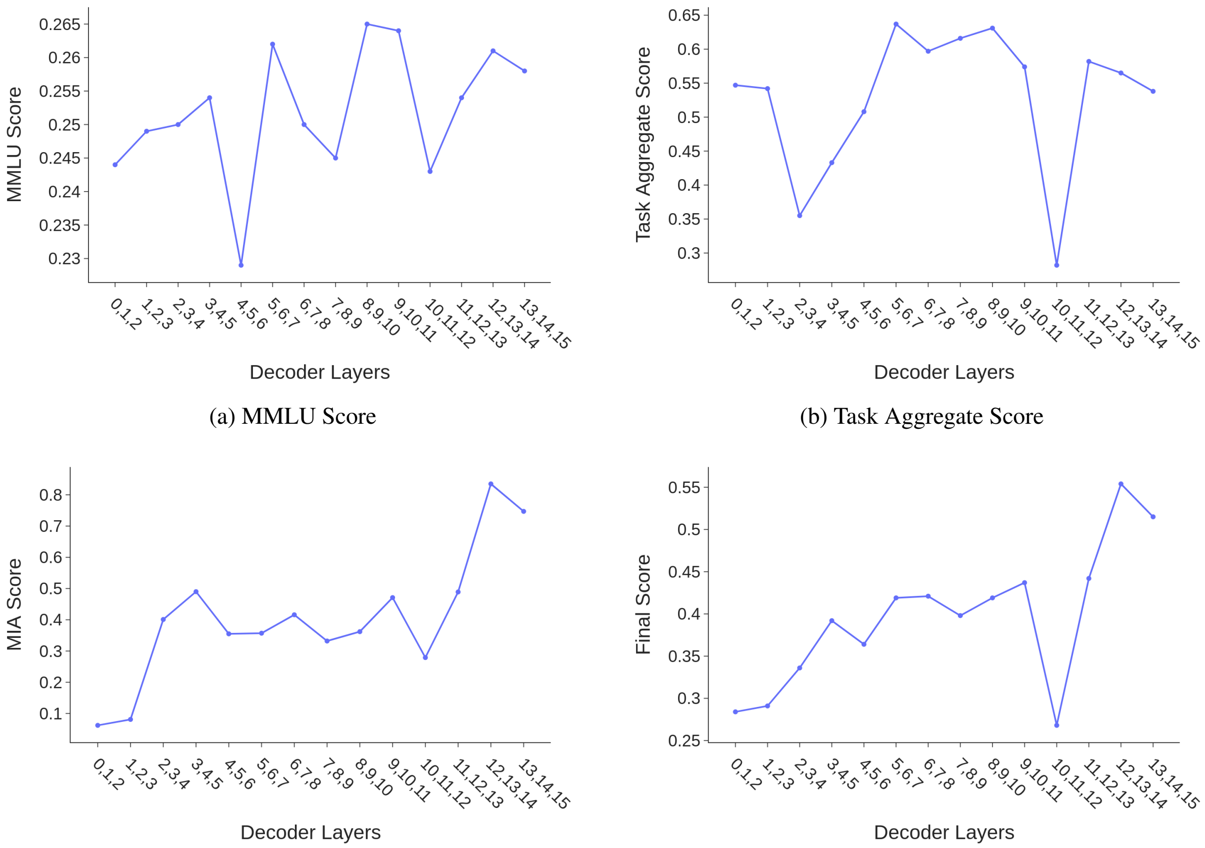
Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. Membership Inference Attacks Against Machine Learning Models . In 2017 IEEE Symposium on Security and Privacy (SP), pages 3–18, Los Alamitos, CA, USA. IEEE Computer Society.

Yunzhi Yao, Peng Wang, Bozhong Tian, Siyuan Cheng, Zhoubo Li, Shumin Deng, Huajun Chen, and Ningyu Zhang. 2023. Editing large language models: Problems, methods, and opportunities. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pages 10222–10240, Singapore. Association for Computational Linguistics.

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. 2024. Negative preference optimization: From catastrophic collapse to effective unlearning.

A Appendix: Analysis of layer combinations for unlearning

Figure 2(a), 2(b) show that unlearning from the middle layers achieves a balance between knowledge retention (high MMLU score) and unlearning performance (high Task Aggregate score), but it remains more susceptible to MIA (Figure 2(c)). In contrast, later layers exhibit significantly higher robustness to MIA. The substantial improvement in MIA robustness outweighs the relatively smaller decline in knowledge retention and unlearning performance, making later layers the more effective choice for unlearning (Figure 2(d)).



(a) MMLU Score

(b) Task Aggregate Score

(c) MIA Score

(d) Final Score

Figure 2: Observed trends of the MMLU Score (a), Task Aggregate Score (b), MIA Score (c), and Final Score (d) vs. unlearned decoder layers.