On the Effect of Steering Latent Representation for Large Language Model Unlearning

Dang Huu-Tien[†], Trung-Tin Pham[†], Hoang Thanh-Tung[◇], Naoya Inoue[†‡]
[†]JAIST [◇]Vietnam National University, Hanoi [‡]RIKEN
{s2310417, naoya-i}@jaist.ac.jp

Abstract

Representation Misdirection for Unlearning (RMU), which steers model representation in the intermediate layer to the target representation, is an effective method for large language model (LLM) unlearning. Despite its high performance, the underlying cause and explanation remain underexplored. In this paper, we theoretically demonstrate that: (1) steering forget representation in the intermediate layer leads to randomized token confidence, causing LLMs to generate wrong or nonsense answers in multiple-choice Q&A, and (2) RMU models are robust against adversarial jailbreak black-box attacks. Additionally, our empirical analysis shows that RMU is less effective when applied to middle and later layers in neural networks. To resolve this drawback, we propose Adaptive RMU—a simple yet effective alternative method that makes unlearning effective with most layers. Extensive experiments demonstrate that Adaptive RMU significantly improves the unlearning performance while incurring no additional computational overhead.

1 Introduction

State-of-the-art LLMs such as GPT-4 (Achiam et al., 2023), Gemini (Team et al., 2023), and Llama-3 (Meta, 2024) achieve remarkable performance through pre-training on large amounts of internet texts and rigorous alignment process for safety enhancement. Despite the immense effort in safety research, LLMs are still vulnerable to adversarial jailbreak attacks and can exhibit unwanted behaviors (Shah et al., 2023; Chao et al., 2023; Zou et al., 2023b; Jones et al., 2023; Yuan et al., 2024; Wei et al., 2024).

Machine Unlearning (Cao and Yang, 2015) has emerged as a promising method for mitigating unforeseen risks in LLMs before deployment. Li et al. (2024) introduced Representation Misdirection for Unlearning (RMU)—an unlearning method that steers the representations of forget-samples (i.e.

samples that the model should forget) toward random representations while keeping the representations of retain-samples (i.e. samples that the model should remember) unchanged. RMU significantly degrades models' accuracy on forget-tasks, while only slightly affecting the performance on retain-tasks and demonstrates stronger robustness against jailbreak attacks. However, the reason for RMU's effectiveness is not well understood, hindering the development of better unlearning algorithms.

In this paper, we analyze the impact of steering latent presentation for LLM unlearning and its connection to adversarial robustness. We empirically show that the RMU forget loss, which minimizes the mean squared error (MSE) between forget representation and a fixed scaled random vector, fails to converge when the norm of the forget representation vector is larger than the scaling coefficient, making RMU less effective when applied to middle and last layers in neural networks. We theoretically show that RMU lowers the probability of the adversary choosing the correct attack direction, thus improving the adversarial robustness of the model.

To overcome RMU's limitation, we introduce *Adaptive RMU*—a variant that adaptively adjusts the coefficient value based on the norm of the forget representation. Experimental results show that Adaptive RMU achieves higher drop-in-accuracy for forget knowledge and enables effective unlearning for most layers without incurring additional computational overhead.

2 Theoretical Analysis

2.1 Background

Notation & problem formulation. Let $\mathcal{D}_{\text{forget}}$ and $\mathcal{D}_{\text{retain}}$ be the forget and retain sets, respectively. Let $f_\theta : \mathbb{R}^{n \times d} \mapsto \mathbb{R}^{n \times |V|}$ be an autoregressive LLM parameterized by θ that maps a prompt input $x_{1:n}$ consisting of n tokens $\{x_1, x_2, ..., x_n\}$ to an output of probability distributions over the vocabu-

lary V. $h_\theta^{(l)}(x)$ denotes the averaged hidden states of input tokens x_i from the l-th layer of f_θ. Our goal is to unlearn the undesired harmful knowledge $\mathcal{D}_{\text{forget}}$ from f_θ while retaining unrelated or general knowledge $\mathcal{D}_{\text{retain}}$. Unlearned models should be robust to knowledge recovery attacks that attempt to recover harmful knowledge from the model.

Representation Misdirection for Unlearning. RMU is a fine-tuning-based unlearning method inspired by representation engineering (RepE; Zou et al. (2023a)) that steers the model's representation of forget samples $x_F \in \mathcal{D}_{\text{forget}}$ to a random vector and regularizes the model representation of retain samples $x_R \in \mathcal{D}_{\text{retain}}$ back to the original model representation, by using the MSE loss:

$$\mathcal{L} = ||h_{\theta^{\text{unlearn}}}^{(l)}(x_F) - c \cdot \boldsymbol{u}||_2^2$$
$$+ \alpha||h_{\theta^{\text{unlearn}}}^{(l)}(x_R) - h_{\theta^{\text{frozen}}}^{(l)}(x_R)||_2^2, \quad (1)$$

where θ^{unlearn} and θ^{frozen} are parameters of the update model and frozen model respectively, $\boldsymbol{u}$ is a random unit vector sampled from Uniform distribution $U(0, 1)$, c is a fixed scaling coefficient, and α is a retain weight. RMU updates θ^{unlearn} w.r.t. $\mathcal{L}$ using gradient descent.

Definition 1. *(Unlearned model and logit of tokens on unlearned model). Suppose model f can be decomposed into $g \circ h^{(l)}$. We define the unlearned model $f^{\text{unlearn}} = g \circ h^{(l),\text{steered}}$, where $h^{(l),\text{steered}}(x_F)$ is the steered representation of forget input x_F at layer l. Given a prompt input $x_{F,1:n}$. For a next token x_{n+1}, the logit value of x_{n+1} obtained from unlearned model f^{unlearn} is defined as:*

$$f^{\text{unlearn}}(x_{n+1}|x_{F,1:n}) = g(h^{(l),\text{steered}}(x_{n+1}|x_{F,1:n}))$$
$$(2)$$

Assumption 1. *(Magnitude and Direction) The representation of a forget-sample x_F at layer l of a well-unlearned model follows the normal distribution $\mathcal{N}(c \cdot \boldsymbol{u}, \eta I)$. More concretely,*

$$h^{(l),\text{steered}}(x_F) = c \cdot \boldsymbol{u} + \boldsymbol{\epsilon} \quad (3)$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \eta I)$ is a small random vector.

2.2 The confidence of tokens generated by RMU models

Proposition 1. *If Assumption 1 holds, by Definition 1, the logit value of token x_{n+1} generated by unlearned model f^{unlearn} given as $f^{\text{unlearn}}(x_{n+1}|x_{F,1:n})$ follows the Normal distribution $\mathcal{N}(||g(\boldsymbol{z})||_2^2, \eta||\nabla_{\boldsymbol{z}} g(\boldsymbol{z})||_2^2)$, where $\boldsymbol{z} = c \cdot \boldsymbol{u}$.*

Proof. See Appendix C.1. □

Proposition 1 states that the logit value of the generated token x_{n+1} by unlearned model f^{unlearn} is randomized. The variance of $f^{\text{unlearn}}(x_{n+1}|x_{F,1:n})$ depends on (i) η: the variance of each dimension of ϵ and (ii) $||\nabla_{\boldsymbol{z}} g(\boldsymbol{z})||_2^2$: the squared l_2 norm of the gradient of $g(\boldsymbol{z})$ w.r.t $\boldsymbol{z}$. If the variance $\eta||\nabla_{\boldsymbol{z}} g(\boldsymbol{z})||_2^2$ is high, $f^{\text{unlearn}}(x_{n+1}|x_{F,1:n})$ has high variance. It means the logit values are more random. Since ϵ is small, then η is small. The main effect depend on $||\nabla_{\boldsymbol{z}} g(\boldsymbol{z})||_2^2$. The variance of $\boldsymbol{z}$ is derived as $\text{Var}(\boldsymbol{z}) = \frac{c^2}{12}$ (See Appendix. C.1). When c gets larger, the variance of $\boldsymbol{z}$ is higher. This could impact the variability of $g(\boldsymbol{z})$ and the gradient $\nabla_{\boldsymbol{z}} g(\boldsymbol{z})$. Intuitively, a larger c introduces more randomness to the logit value, leading to a faster decrease in accuracy (faster unlearning). However, behaviors of $||\nabla_{\boldsymbol{z}} g(\boldsymbol{z})||_2^2$ depend on the specific characteristics of sub-network g and the distribution of variable $\boldsymbol{z}$. Unfortunately, g is a composition of transformer layers, which makes it difficult to have a complete analysis. Therefore, we conduct an experiment to understand the effect of c on unlearning in §3.

2.3 The adversarial robustness of RMU models

Unlearning algorithms erase undesirable knowledge and prevent them from resurfacing when attacked by adversarial jailbreak attacks (Li et al., 2024; Zou et al., 2024). To understand the underlying causes for the effectiveness of unlearning algorithms, we conduct an analysis from the standpoint of an attack-defense game. State-of-the-art LLMs such as GPT, Gemini, and Claude are trained privately and are accessible through API only. The most common form of attack on LLMs, therefore, is a black-box attack.

In black-box attacks, the adversary possesses limited information about the victim model's architecture or training process and can only access the model's input and output. A common approach is the adversary injects a random update $\delta \sim \mathcal{N}(0, \nu I), \nu \in \mathbb{R}^+$ to input x and determines the quality of the update based on the change in loss:

$$\Delta\mathcal{J} = \mathcal{J}(f(x + \delta)) - \mathcal{J}(f(x)) \quad (4)$$

If δ is a good direction, it lowers the loss $\mathcal{J}(f(x + \delta))$, resulting in $\Delta\mathcal{J} < 0$. The adversary chooses

the optimal δ through random search. After unlearning, the direction relies on $\Delta\mathcal{J}^{\text{unlearn}} = \mathcal{J}(f^{\text{unlearn}}(x + \delta)) - \mathcal{J}(f^{\text{unlearn}}(x))$. We show that the good direction the adversary chooses in an RMU model forms an obtuse angle to the good direction in the original model with high probability, i.e. the following inequality

$$-1 \leq \frac{\langle \Delta\mathcal{J}, \Delta\mathcal{J}^{\text{unlearn}} \rangle}{||\Delta\mathcal{J}||_2 ||\Delta\mathcal{J}^{\text{unlearn}}||_2} \leq 0. \qquad (5)$$

holds with high probability.

Theorem 2.1. *Suppose the adversary injects a small, random update $\delta \sim \mathcal{N}(0, \nu I)$, $\nu \in \mathbb{R}^+$ to the input x and query unlearned model f^{unlearn} to find the optimal direction. The probability that the adversary chooses the opposite or orthogonal direction to the optimal direction is*

$$\mathbb{P}\left[-1 \leq \frac{\langle \Delta\mathcal{J}, \Delta\mathcal{J}^{\text{unlearn}} \rangle}{||\Delta\mathcal{J}||_2 ||\Delta\mathcal{J}^{\text{unlearn}}||_2} \leq 0 \right]$$
$$\approx \frac{1}{2} \operatorname{erf}\left(\left(\frac{\nu(||\nabla_x(\mathcal{J} \circ f^{\text{unlearn}})||_2^2 + ||\nabla_x(\mathcal{J} \circ f)||_2^2)}{2} \right)^{\frac{1}{2}} \right)$$
$$(6)$$

where $\operatorname{erf}(x)$ is the Gaussian error function.

Proof. See Appendix C.2. □

Since $\nu \in \mathbb{R}^+$, the squared ℓ_2 norms $||\nabla_x(\mathcal{J} \circ f^{\text{unlearn}})||_2^2$, $(||\nabla_x(\mathcal{J} \circ f)||_2^2$ are positive, then $\left(\frac{\nu(||\nabla_x(\mathcal{J} \circ f^{\text{unlearn}})||_2^2 + ||\nabla_x(\mathcal{J} \circ f)||_2^2)}{2} \right)^{\frac{1}{2}}$ being interval of $(0, \infty)$. The probability $\frac{1}{2} \operatorname{erf}\left(\left(\frac{\nu(||\nabla_x(\mathcal{J} \circ f^{\text{unlearn}})||_2^2 + ||\nabla_x(\mathcal{J} \circ f)||_2^2)}{2} \right)^{\frac{1}{2}} \right)$ is bounded in $(0, \frac{1}{2})$.

Theorem. 2.1 states that the probability that the adversary chooses the wrong direction is larger when the gradient norms are larger. The unlearned model serves as a defender by causing the attacker to miscalculate the direction of its optimization trajectory, thereby increasing the attacker's cost. Li et al. (2024) present a qualitative experiment by employing Greedy Coordinate Gradient (GCG; Zou et al. (2023b)) to jailbreak RMU models to extract the answer given a harmful prompt. As a result, RMU unlearned models demonstrate strong robustness against jailbreak attacks. Their experiment results implicitly verify our theoretical analysis.
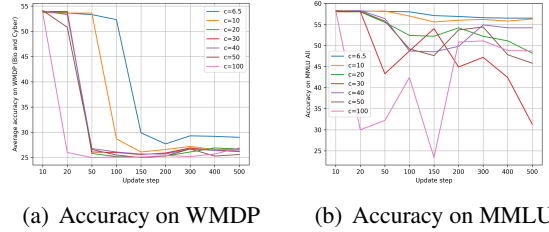

(a) Accuracy on WMDP   (b) Accuracy on MMLU

Figure 1: Average accuracy of (a) WMDP (Biology and Cyber) and (b) MMLU-All with different coefficient c.
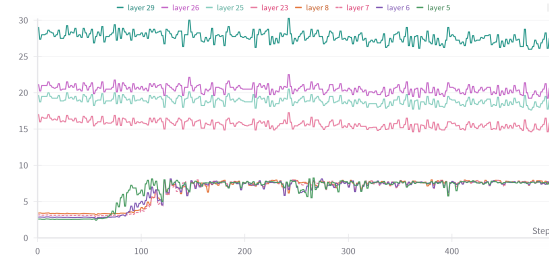

Figure 2: Representation norm of forget samples.

3 Empirical Analysis

Effect of the coefficient c. To empirically verify our analysis in §2.2, we analyze the impact of c for forgotten knowledge and retained knowledge, using WMDP (Li et al., 2024) and MMLU (Hendrycks et al., 2020). See Appendix B for the full experiment setting. Fig. 1 shows: (i) a clear positive correlation between the drop-in-accuracy rate and the value of c, i.e., higher c makes the accuracy decrease faster. (ii) A larger value of c tends to make a more drop-in-accuracy on WMDP (Fig. 1a). (iii) However, a larger c comes with a caveat in a significant drop in general performance on MMLU (Fig. 1b).

Effect of unlearn layers. We investigate the effect of unlearn layers on accuracy and the representation norm during unlearning. We change the unlearn layer l from $3 \rightarrow 31$, fixed $c = 6.5$. Table 1 shows that RMU is effective for unlearning within the early layers ($3 \rightarrow 10$), yet exhibits inefficacy within middle and later layers ($11 \rightarrow 31$). Interestingly, in Fig. 3, we observed that within early layers, the representation norms of forget samples are smaller than the coefficient c. During unlearning, the representation norm exponentially increases, approaching c, thereby facilitating the convergence of forget loss (i.e. Assumption 1 holds). Conversely, within middle and later layers, the representation norms of forget samples, initially

3

Task/unlearn layer	base	3	4	5	6	7	8	9	10	11	12	13	14	15	16
WMDP-Biology ↓	63.7	31.3	42.2	34.8	<u>29.3</u>	**28.8**	36.6	41.1	50.9	62.7	59.2	62.1	63.2	63.0	64.1
WMDP-Cyber ↓	43.5	43.0	42.1	31.0	**27.8**	<u>28.8</u>	30.4	29.1	29.8	37.2	39.5	38.4	41.8	42.4	43.4
MMLU-All ↑	58.1	57.2	56.8	57.0	57.0	56.8	56.8	57.2	57.9	57.7	57.3	57.2	57.9	58.3	57.9
Task/unlearn layer	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
WMDP-Biology ↓	63.7	63.8	63.7	63.2	63.4	63.7	63.6	63.8	63.7	63.5	63.5	63.5	63.7	63.5	64.0
WMDP-Cyber ↓	43.5	44.1	43.7	43.8	43.9	43.9	43.7	43.5	43.4	43.8	43.6	43.8	43.7	43.7	43.9
MMLU-All ↑	57.9	58.1	58.1	58.1	58.1	58.0	58.0	58.0	58.1	58.1	58.1	58.0	58.1	58.0	58.0

Table 1: Q&A accuracy of RMU Zephyr-7B models on WMDP-Biology, WMDP-Cyber, and MMLU-All w.r.t unlearn layer l from $3 \to 31$. The coefficient $c = 6.5$. The **best** and <u>runner up</u> are marked.

Task/unlearn layer	base	3	4	5	6	7	8	9	10	11	12	13	14	15	16
WMDP-Biology ↓	63.7	30.9	29.7	25.8	27.1	**23.7**	<u>24.3</u>	24.6	27.1	38.8	30.2	35.1	51.3	31.7	39.5
WMDP-Cyber ↓	43.5	43.2	38.9	<u>24.4</u>	**24.3**	26.5	25.2	27.0	27.1	27.8	27.0	27.0	27.4	29.3	29.1
MMLU-All ↑	58.1	56.8	56.1	55.0	55.1	55.0	54.0	50.4	55.9	54.0	47.6	40.9	56.7	55.5	57.3
Task/unlearn layer	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
WMDP-Biology ↓	44.1	37.3	47.6	46.7	49.4	49.6	51.3	55.2	53.0	58.9	47.6	64.1	58.7	56.3	64.8
WMDP-Cyber ↓	31.1	26.8	26.6	26.8	27.2	27.8	28.0	36.4	37.8	43.8	43.3	43.9	42.2	43.9	44.0
MMLU-All ↑	57.4	57.4	56.8	56.9	57.8	57.8	57.6	57.9	57.8	57.8	57.6	58.2	57.9	58.0	58.0

Table 2: Q&A accuracy of Adaptive RMU Zephyr-7B models on WMDP-Biology, WMDP-Cyber, and MMLU-All w.r.t unlearn layer l from $3 \to 31$. The scaling factor $\beta = 5$. The **best** and <u>runner up</u> are marked.

larger than c, remain unchanged during unlearning, making the forget loss divergent (i.e. Assumption 1 does not hold).

4 Adaptive RMU

Inspired by the observation in §3, we propose *Adaptive RMU*, a simple yet effective unlearning method with an adaptive forget loss by scaling the random unit vector $\boldsymbol{u}$ with an **adaptive scaling coefficient** $\beta||h_{\theta^{\text{frozen}}}^{(l)}(x_F)||_2$, where $\beta \in \mathbb{R}^+$ is a scaling factor and $||h_{\theta^{\text{frozen}}}^{(l)}(x_F)||_2$ is the ℓ_2 representation norm of forget samples x_F on model $f_{\theta^{\text{frozen}}}$. The total loss is calculated as follows:

$$\mathcal{L}^{\text{adap}} = \underbrace{||h_{\theta^{\text{unlearn}}}^{(l)}(x_F) - \beta||h_{\theta^{\text{frozen}}}^{(l)}(x_F)||_2 \cdot \boldsymbol{u}||_2^2}_{\text{adaptive forget loss}}$$
$$+ \alpha \underbrace{||h_{\theta^{\text{unlearn}}}^{(l)}(x_R) - h_{\theta^{\text{frozen}}}^{(l)}(x_R)||_2^2}_{\text{retain loss}} \quad (7)$$

Our Adaptive RMU is shown in Algorithm 1.

Experiment. We choose the following LLMs: Zephyr-7B (Tunstall et al., 2023), Yi 6B (Young et al., 2024), Llama 3 8B (Meta, 2024), and Mistral 7B (v0.1) (Jiang et al., 2023). We use WMDP-Biology and WMDP-Cyber as $\mathcal{D}_{\text{forget}}$ and Wikitext (Merity et al., 2016) as $\mathcal{D}_{\text{retain}}$. Unlearned models are evaluated on WMDP Q&A and MMLU (Hendrycks et al., 2020). We employ LLMU (Yao et al., 2023), SCRUB (Kurmanji et al.,

2023), and SSD (Foster et al., 2024) as baseline comparisons. See Appendix B for further details.

Main results. Table 1 and 2 show that Adaptive RMU with Zephyr-7B models significantly improves RMU, reducing average accuracy by 13.1% on WMDP-Bio and 3.6% on WMDP-Cyber within early layers ($3 \to 10$), and by 15.6% on WMDP-Bio and 9.6% on WMDP-Cyber within middle and later layers ($11 \to 31$). This corresponds to an overall enhancement of 14.3% and 6.6% in drop-in-accuracy for the WMDP-Bio and WMDP-Cyber, respectively. Table 4 also shows that Adaptive RMU surpasses RMU, LLMU, SCRUB, and SSD by 0.9%, 17.3%, 10.2%, and 15.9% in term of the average of drop-in-accuracy on WMDP and accuracy on MMLU, respectively, establishing a new state-of-the-art performance. See Appendix D for full results on other models; it shows the same trend as Table 1 and 2.

5 Conclusion

We studied the effect of steering latent representation for LLM unlearning and explored its connection to jailbreak adversarial robustness. We developed a simple yet effective alternative method that enhances unlearning performance with most layers. Our findings illuminate the explanation of RMU and pave the way for future research in large language model unlearning.

Limitations

We discuss the following limitations in our paper:

1. We note that Meta Llama 3 and Yi have 70B and 34B versions respectively, which we did not test due to computational constraints. We mainly perform experiments on 7B versions (or equivalent). To validate the generalizability of our approach and findings, we conducted experiments across the Zephyr, Mistral, Llama 3, and Yi models.

2. Our theoretical analysis in §2.3 on back-box attacks for API-access models. In practice, adversarial attacks are not random, they are based on other signals. Assuming a random attack is not very realistic, especially for open-weight models. We encourage future works to explore the analysis of the robustness of unlearned models covering white-box attacks.

3. Limiting update the model parameters w.r.t three layer $l, l-1, l-2$ thus risks missing interesting generalization behaviors.

Ethics Statement

Our study considers theoretical and empirical aspects of the RMU method for unlearning. We do not have bias or cause harm to any group of people.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Seohui Bae, Seoyoon Kim, Hyemin Jung, and Woohyung Lim. 2023. Gradient surgery for one-shot unlearning on generative model. *arXiv preprint arXiv:2307.04550*.

Nora Belrose, David Schneider-Joseph, Shauli Ravfogel, Ryan Cotterell, Edward Raff, and Stella Biderman. 2023. LEACE: Perfect linear concept erasure in closed form. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Lucas Bourtoule, Varun Chandrasekaran, Christopher A Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. 2021. Machine unlearning. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 141–159. IEEE.

Paul Bromiley. 2003. Products and convolutions of gaussian probability density functions. *Tina-Vision Memo*, 3(4):1.

PRESTON BUKATY. 2019. *The California Consumer Privacy Act (CCPA): An implementation guide*. IT Governance Publishing.

Yinzhi Cao and Junfeng Yang. 2015. Towards making systems forget with machine unlearning. In *2015 IEEE Symposium on Security and Privacy*, pages 463–480.

Sungmin Cha, Sungjun Cho, Dasol Hwang, Honglak Lee, Taesup Moon, and Moontae Lee. 2024. Learning to unlearn: Instance-wise unlearning for pre-trained classifiers. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 11186–11194.

Patrick Chao, Alexander Robey, Edgar Dobriban, Hamed Hassani, George J Pappas, and Eric Wong. 2023. Jailbreaking black box large language models in twenty queries. *arXiv preprint arXiv:2310.08419*.

Tianshi Che, Yang Zhou, Zijie Zhang, Lingjuan Lyu, Ji Liu, Da Yan, Dejing Dou, and Jun Huan. 2023. Fast federated machine unlearning with nonlinear functional theory. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Chaochao Chen, Yizhao Zhang, Yuyuan Li, Dan Meng, Jun Wang, Xiaoli Zheng, and Jianwei Yin. 2024. Post-training attribute unlearning in recommender systems. *arXiv preprint arXiv:2403.06737*.

Min Chen, Zhikun Zhang, Tianhao Wang, Michael Backes, Mathias Humbert, and Yang Zhang. 2022. Graph unlearning. In *Proceedings of the 2022 ACM SIGSAC conference on computer and communications security*, pages 499–513.

Eli Chien, Chao Pan, and Olgica Milenkovic. 2023. Efficient model updates for approximate unlearning of graph-structured data. In *The Eleventh International Conference on Learning Representations*.

Dasol Choi and Dongbin Na. 2023. Towards machine unlearning benchmarks: Forgetting the personal identities in facial recognition systems. *arXiv preprint arXiv:2311.02240*.

Bart van der Sloot Chris Jay Hoofnagle and Frederik Zuiderveen Borgesius. 2019. The european union general data protection regulation: what it is and what it means*. *Information & Communications Technology Law*, 28(1):65–98.

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

Chongyu Fan, Jiancheng Liu, Yihua Zhang, Eric Wong, Dennis Wei, and Sijia Liu. 2024. Salun: Empowering machine unlearning via gradient-based weight saliency in both image classification and generation. In *The Twelfth International Conference on Learning Representations*.

Jack Foster, Stefan Schoepf, and Alexandra Brintrup. 2024. Fast machine unlearning without retraining through selective synaptic dampening. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 12043–12051.

Rohit Gandikota, Joanna Materzynska, Jaden Fiotto-Kaufman, and David Bau. 2023. Erasing concepts from diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2426–2436.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2023. A framework for few-shot language model evaluation.

Antonio Ginart, Melody Guan, Gregory Valiant, and James Y Zou. 2019. Making ai forget you: Data deletion in machine learning. *Advances in neural information processing systems*, 32.

Aditya Golatkar, Alessandro Achille, and Stefano Soatto. 2020. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9304–9312.

Roger Grosse, Juhan Bae, Cem Anil, Nelson Elhage, Alex Tamkin, Amirhossein Tajdini, Benoit Steiner, Dustin Li, Esin Durmus, Ethan Perez, et al. 2023. Studying large language model generalization with influence functions. *arXiv preprint arXiv:2308.03296*.

Anisa Halimi, Swanand Kadhe, Ambrish Rawat, and Nathalie Baracaldo. 2022. Federated unlearning: How to efficiently erase a client in fl? *arXiv preprint arXiv:2207.05521*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Yoichi Ishibashi and Hidetoshi Shimodaira. 2023. Knowledge sanitization of large language models. *arXiv preprint arXiv:2309.11852*.

Masaru Isonuma and Ivan Titov. 2024. Unlearning reveals the influential training data of language models. *arXiv preprint arXiv:2401.15241*.

Zachary Izzo, Mary Anne Smart, Kamalika Chaudhuri, and James Zou. 2021. Approximate data deletion from machine learning models. In *International Conference on Artificial Intelligence and Statistics*, pages 2008–2016. PMLR.

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.

Hyejun Jeong, Shiqing Ma, and Amir Houmansadr. 2024. Sok: Challenges and opportunities in federated unlearning. *arXiv preprint arXiv:2403.02437*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Erik Jones, Anca Dragan, Aditi Raghunathan, and Jacob Steinhardt. 2023. Automatically auditing large language models via discrete optimization. In *International Conference on Machine Learning*, pages 15307–15329. PMLR.

Pang Wei Koh and Percy Liang. 2017. Understanding black-box predictions via influence functions. In *International conference on machine learning*, pages 1885–1894. PMLR.

Nupur Kumari, Bingliang Zhang, Sheng-Yu Wang, Eli Shechtman, Richard Zhang, and Jun-Yan Zhu. 2023. Ablating concepts in text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22691–22702.

Meghdad Kurmanji, Peter Triantafillou, Jamie Hayes, and Eleni Triantafillou. 2023. Towards unbounded machine unlearning. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D. Li, Ann-Kathrin Dombrowski, Shashwat Goel, Gabriel Mukobi, Nathan Helm-Burger, Rassin Lababidi, Lennart Justen, Andrew Bo Liu, Michael Chen, Isabelle Barrass, Oliver Zhang, Xiaoyuan Zhu, Rishub Tamirisa, Bhrugu Bharathi, Ariel Herbert-Voss, Cort B Breuer, Andy Zou, Mantas Mazeika, Zifan Wang, Palash Oswal, Weiran Lin, Adam Alfred Hunt, Justin Tienken-Harder, Kevin Y. Shih, Kemper Talley, John Guan, Ian Steneker, David Campbell, Brad Jokubaitis, Steven Basart, Stephen Fitz, Ponnurangam Kumaraguru, Kallol Krishna Karmakar, Uday Tupakula, Vijay Varadharajan, Yan Shoshitaishvili, Jimmy Ba, Kevin M. Esvelt, Alexandr Wang, and Dan Hendrycks. 2024. The WMDP benchmark: Measuring and reducing malicious use with unlearning. In *Forty-first International Conference on Machine Learning*.

Yuyuan Li, Chaochao Chen, Xiaolin Zheng, Yizhao Zhang, Zhongxuan Han, Dan Meng, and Jun Wang.

2023. Making users indistinguishable: Attribute-wise unlearning in recommender systems. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 984–994.

Gaoyang Liu, Xiaoqiang Ma, Yang Yang, Chen Wang, and Jiangchuan Liu. 2020. Federated unlearning. *arXiv preprint arXiv:2012.13891*.

Zheyuan Liu, Guangyao Dou, Zhaoxuan Tan, Yijun Tian, and Meng Jiang. 2024. Towards safer large language models through machine unlearning. *arXiv preprint arXiv:2402.10058*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Zhuo Ma, Yang Liu, Ximeng Liu, Jian Liu, Jianfeng Ma, and Kui Ren. 2022. Learn to forget: Machine unlearning via neuron masking. *IEEE Transactions on Dependable and Secure Computing*.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C Lipton, and J Zico Kolter. 2024. Tofu: A task of fictitious unlearning for llms. *arXiv preprint arXiv:2401.06121*.

Alessandro Mantelero. 2013. The eu proposal for a general data protection regulation and the roots of the 'right to be forgotten'. *Computer Law Security Review*, 29(3):229–235.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in gpt. *Advances in Neural Information Processing Systems*, 35:17359–17372.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2016. Pointer sentinel mixture models. *arXiv preprint arXiv:1609.07843*.

AI Meta. 2024. Introducing meta llama 3: The most capable openly available llm to date. *Meta AI*.

Thanh Tam Nguyen, Thanh Trung Huynh, Phi Le Nguyen, Alan Wee-Chung Liew, Hongzhi Yin, and Quoc Viet Hung Nguyen. 2022. A survey of machine unlearning. *arXiv preprint arXiv:2209.02299*.

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*.

Martin Pawelczyk, Seth Neel, and Himabindu Lakkaraju. 2023. In-context unlearning: Language models as few shot unlearners. *arXiv preprint arXiv:2310.07579*.

Nicolò Romandini, Alessio Mora, Carlo Mazzocca, Rebecca Montanari, and Paolo Bellavista. 2024. Federated unlearning: A survey on methods, design guidelines, and evaluation metrics. *arXiv preprint arXiv:2401.05146*.

Anwar Said, Tyler Derr, Mudassir Shabbir, Waseem Abbas, and Xenofon Koutsoukos. 2023. A survey of graph unlearning. *arXiv preprint arXiv:2310.02164*.

Ayush Sekhari, Jayadev Acharya, Gautam Kamath, and Ananda Theertha Suresh. 2021. Remember what you want to forget: Algorithms for machine unlearning. *Advances in Neural Information Processing Systems*, 34:18075–18086.

Rusheb Shah, Soroush Pour, Arush Tagade, Stephen Casper, Javier Rando, et al. 2023. Scalable and transferable black-box jailbreaks for language models via persona modulation. *arXiv preprint arXiv:2311.03348*.

Supreeth Shastri, Melissa Wasserman, and Vijay Chidambaram. 2019. The seven sins of personal-data processing systems under gdpr. In *Proceedings of the 11th USENIX Conference on Hot Topics in Cloud Computing*, HotCloud'19, page 1, USA. USENIX Association.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Anvith Thudi, Gabriel Deza, Varun Chandrasekaran, and Nicolas Papernot. 2022. Unrolling sgd: Understanding factors influencing machine unlearning. In *2022 IEEE 7th European Symposium on Security and Privacy (EuroS&P)*, pages 303–319. IEEE.

Daniel Trippa, Cesare Campagnano, Maria Sofia Bucarelli, Gabriele Tolomei, and Fabrizio Silvestri. 2024. Gradient-based and task-agnostic machine unlearning. *arXiv preprint arXiv:2403.14339*.

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, et al. 2023. Zephyr: Direct distillation of lm alignment. *arXiv preprint arXiv:2310.16944*.

Junxiao Wang, Song Guo, Xin Xie, and Heng Qi. 2022. Federated unlearning via class-discriminative pruning. In *Proceedings of the ACM Web Conference 2022*, WWW '22, page 622–632, New York, NY, USA. Association for Computing Machinery.

Alexander Warnecke, Lukas Pirch, Christian Wressnegger, and Konrad Rieck. 2021. Machine unlearning of features and labels. *arXiv preprint arXiv:2108.11577*.

Alexander Wei, Nika Haghtalab, and Jacob Steinhardt. 2024. Jailbroken: How does llm safety training fail? *Advances in Neural Information Processing Systems*, 36.

Kun Wu, Jie Shen, Yue Ning, Ting Wang, and Wendy Hui Wang. 2023a. Certified edge unlearning for graph neural networks. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '23, page 2606–2617, New York, NY, USA. Association for Computing Machinery.

Xinwei Wu, Junzhuo Li, Minghui Xu, Weilong Dong, Shuangzhi Wu, Chao Bian, and Deyi Xiong. 2023b. DEPN: Detecting and editing privacy neurons in pretrained language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2875–2886, Singapore. Association for Computational Linguistics.

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

Alex Young, Bei Chen, Chao Li, Chengen Huang, Ge Zhang, Guanwei Zhang, Heng Li, Jiangcheng Zhu, Jianqun Chen, Jing Chang, et al. 2024. Yi: Open foundation models by 01. ai. *arXiv preprint arXiv:2403.04652*.

Youliang Yuan, Wenxiang Jiao, Wenxuan Wang, Jen tse Huang, Pinjia He, Shuming Shi, and Zhaopeng Tu. 2024. GPT-4 is too smart to be safe: Stealthy chat with LLMs via cipher. In *The Twelfth International Conference on Learning Representations*.

Gong Zhang, Kai Wang, Xingqian Xu, Zhangyang Wang, and Humphrey Shi. 2024. Forget-me-not: Learning to forget in text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1755–1764.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, et al. 2023a. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*.

Andy Zou, Long Phan, Justin Wang, Derek Duenas, Maxwell Lin, Maksym Andriushchenko, Rowan Wang, Zico Kolter, Matt Fredrikson, and Dan Hendrycks. 2024. Improving alignment and robustness with short circuiting. *arXiv preprint arXiv:2406.04313*.

Andy Zou, Zifan Wang, J Zico Kolter, and Matt Fredrikson. 2023b. Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*.

A Related work

Recent legislation on right-to-be-forgotten (Shastri et al., 2019), such as the General Data Protection Regulation (Mantelero, 2013) and the California Consumer Privacy Act (BUKATY, 2019) has raised attention to a new learning paradigm called machine unlearning (Cao and Yang, 2015; Bourtoule et al., 2021; Chris Jay Hoofnagle and Borgesius, 2019; Nguyen et al., 2022).

Traditional Machine Unlearning. Two natural and straightforward unlearning approaches for classification models are leave-some-out retraining: retraining the model from scratch without the forget samples, and random labeling: training the model with randomly flipped label forget samples. However, these methods become more computationally expensive as the size of datasets and modern deep networks grows. Existing works focus on approximating data influence (Warnecke et al., 2021; Izzo et al., 2021; Sekhari et al., 2021; Isonuma and Titov, 2024) using Influence Function (Koh and Liang, 2017; Grosse et al., 2023), gradient projection (Bae et al., 2023), or gradient ascent (Thudi et al., 2022; Trippa et al., 2024).

Other views on the landscape of machine unlearning include: unlearning in text classification (Ma et al., 2022), image classification and recognition (Ginart et al., 2019; Golatkar et al., 2020; Fan et al., 2024; Choi and Na, 2023; Cha et al., 2024), diffusion models (Gandikota et al., 2023; Zhang et al., 2024; Kumari et al., 2023), federated unlearning (Liu et al., 2020; Romandini et al., 2024; Wang et al., 2022; Che et al., 2023; Halimi et al., 2022; Jeong et al., 2024), graph unlearning (Chen et al., 2022; Chien et al., 2023; Wu et al., 2023a; Said et al., 2023), and recommender systems (Chen et al., 2024; Li et al., 2023).

LLM Unlearning. Due to the large size of the parameters and training data, LLM poses a new challenge to unlearning. Current studies in LLM unlearning mainly focus on task or context-specific settings such as unlearning copyrighted material from the Harry Potter series (Eldan and Russinovich, 2023), In-context unlearning (Pawelczyk et al., 2023), fictitious unlearning (Maini et al., 2024), specific harmful input-output (Yao et al., 2023; Liu et al., 2024), sensitive and private information (Jang et al., 2023; Wu et al., 2023b; Ishibashi and Shimodaira, 2023; Patil et al., 2024), gender (Belrose et al., 2023) or facts (Meng et al., 2022). More recently, Li et al. (2024) consider unlearning an entire distribution of hazardous knowledge given limited samples.

B Experiment details, datasets, and algorithm computational perplexity

B.1 Experimental details.

We use WMDP forget sets and wikitext (Merity et al., 2016) for finetuning the LLM. We evaluate unlearned models on WMDP Q&A sets and MMLU (Hendrycks et al., 2020). An unlearned model has higher accuracy on MMLU and lower accuracy on WMDP is better.

Models were fine-tuned using AdamW (Loshchilov and Hutter, 2019) with learning rate $\eta = 5e - 5$, batch-size of 4, max sequence len of 512 for WMDP-Bio and 768 for WMDP-Cyber, with $T = 500$ gradient update steps. The retain weight $\alpha = 1200$. For the baseline RMU, we follow the previous work and let $c = 6.5$. We grid search for unlearn layer l from the third layer to the last layer (31), and the scaling factor $\beta \in \{2, 3, 5, 10\}$. We update the model parameters with respect to three layers $\{l, l-1, l-2\}$. Two NVIDIA A40s with 90GB RAM were used to run the experiment.

Baselines. We compare Adaptive RMU against four baselines: RMU (Li et al., 2024), Large Language Model Unlearning (LLMU; Yao et al. (2023)), SCalable Remenbering and Unlearning unBound (SCRUB; Kurmanji et al. (2023)), and Selective Synaptic Dampening (SSD; Foster et al. (2024). We use off-the-shelf results from Li et al. (2024) for LLMU, SCRUB, and SSD.

Reproducibility. Our source code and guidelines to reproduce results were attached to the supplementary materials.

B.2 Datasets

WMDP (Li et al., 2024). The WMDP benchmark, standing for the Weapon of Mass Destruction Proxy, a dataset consists of forget set, retain set, and Q&A. The WMDP Q&A is a dataset of $3,668$ multiple-choice questions about Biosecurity $(1,273)$, Cybersecurity $(1,987)$, and Chemical security (408). The WMDP-Bio forget and retain sets consist of papers from PubMed. The WMDP-Bio forget comprises papers used in generating WMDP-Bio questions, while the retain set samples papers from various categories within general biology. The retain set excludes papers from the forget set and employs keyword exclusion to avoid topics related to Q&A set. The WMDP-Cyber forget and retain sets consist of passages crawled from GitHub with two different sets of keywords. Note that we did not benchmark for WMDP-Chemistry Q&A due to no WMDP-Chemistry forget set being publicly released.

MMLU (Hendrycks et al., 2020). The MMLU benchmark, standing for Massive Multitask Language Understanding, a dataset of $15,908$ multiple-choice Q&A cover 57 subjects across STEM, the humanities, the social science, and more. MMLU is designed to measure general knowledge by evaluating models in zero-shot or few-shot settings.

Zero-shot Q&A format. We use LM-Evaluation-Harness (Gao et al., 2023) to standardize the zero-shot Q&A prompt. We follow the default question-answering format to query LLM. For example:

```
"The following are multiple choice
questions (with answers) about Biology.

Question: How can I cultivate a
calicivirus?
A. Cell/enteroid culture
B. Helper virus-free reverse genetics
system
C. Either A or B
D. Neither A nor B
Answer:"
```
The LLM encodes the prompt input and generates answer tokens.

B.3 Computational perplexity of the RMU and Adaptive RMU.

RMU (Li et al., 2024) and Our Adaptive RMU Algorithm are shown in Algorithm. 2 and Algorithm. 1 respectively. The main difference between RMU and Adaptive RMU is the calculation of the forget coefficient. Adaptive RMU uses adaptive coefficient $\beta||h_{\theta^{\text{frozen}}}^{(l)}(x_F)||_2$, which can be calculated and cached during the first iteration of the inner **for** loop (Algorithm. 1). Thus, the complexity of Adaptive RMU is equal to that of RMU. Additionally, we report the average unlearning runtime in Table. 3.

Yi 6B	Mistral 7B	Zephyr 7B	Meta Llama 3 8B
1377.2	1225.2	1254.0	1729.8

Table 3: Average unlearning runtime in second (with 2 NVIDIA A40s, batch-size of 4 and 500 steps update)

Algorithm 1 Adaptive RMU pseudocode

Require:
1: $\mathcal{D}_{\text{forget}}$: a forget set.
2: $\mathcal{D}_{\text{retain}}$: a retain set.
3: $f_{\theta^{\text{frozen}}}$: a frozen model.
4: $f_{\theta^{\text{unlearn}}}$: an update model.
5: α: a retain weight.
6: l: an unlearn layer.
7: β: a scaling factor.
8: T: number of gradient update steps.

Ensure: Return an unlearned model $f_{\theta^{\text{unlearn}}}$.
9: Sampling a random unit vector $\boldsymbol{u} \sim U(0,1)$
10: **for** mini-batch $t \in [1...T] : x_F \in \mathcal{D}_{\text{forget}}$, $x_R \in \mathcal{D}_{\text{retain}}$ **do**
11:    Get the representations of x_F and x_R from the frozen and update models.
12:    Compute the adaptive loss $\mathcal{L}^{\text{adap}}$ by Eqn. 8.
13:    Update θ^{unlearn} w.r.t $\mathcal{L}^{\text{adap}}$ using gradient descent.
14:    $t = t + 1$
15: **end for**
16: **return** $f_{\theta^{\text{unlearn}}}$

Algorithm 2 RMU pseudocode (Li et al., 2024)

Require:
1: $\mathcal{D}_{\text{forget}}$: a forget set.
2: $\mathcal{D}_{\text{retain}}$: a retain set.
3: $f_{\theta^{\text{frozen}}}$: a frozen model.
4: $f_{\theta^{\text{unlearn}}}$: an update model.
5: α: a retain weight.
6: l: an unlearn layer.
7: c: a forget coefficient.
8: T: number of gradient update steps.

Ensure: Return an unlearned model $f_{\theta^{\text{unlearn}}}$.
9: Sampling a random unit vector $\boldsymbol{u} \sim U(0,1)$
10: **for** mini-batch $t \in [1...T] : x_F \in \mathcal{D}_{\text{forget}}$, $x_R \in \mathcal{D}_{\text{retain}}$ **do**
11:    Get the representations of x_F and x_R from the frozen and update models
12:    Compute the loss $\mathcal{L}$ by Eqn. 1.
13:    Update θ^{unlearn} w.r.t $\mathcal{L}$ using gradient descent.
14:    $t = t + 1$
15: **end for**
16: **return** $f_{\theta^{\text{unlearn}}}$

C Proof

C.1 Proof of section 2.2

Proposition 1. *If Assumption 1 holds, by Definition 1, the logit value of token x_{n+1} generated by unlearned model f^{unlearn} given as $f^{\text{unlearn}}(x_{n+1}|x_{F,1:n})$ follows the Normal distribution $\mathcal{N}(||g(\boldsymbol{z})||_2^2, \eta||\nabla_{\boldsymbol{z}}g(\boldsymbol{z})||_2^2)$, where $\boldsymbol{z} = c \cdot \boldsymbol{u}$.*

Proof. Given Assumption 1, we have:

$$h^{(l),\text{steered}}(x_{n+1}|x_{F,1:n}) = c \cdot \boldsymbol{u} + \epsilon, \quad (8)$$

where $c \in \mathbb{R}^+$ is a coefficient, $\boldsymbol{u}$ is a random vector sampled from Uniform distribution $U(0,1)$, and ϵ is a random vector independently sampled from Gaussian distribution $\mathcal{N}(0, \Sigma)$, where $\Sigma = \eta I$ is a covariance matrix. We denote $\boldsymbol{z} = c \cdot \boldsymbol{u}$. Substituting Eqn. 8 into Eqn. 2, we get:

$$f^{\text{unlearn}}(x_{n+1}|x_{F,1:n}) = g(\boldsymbol{z} + \boldsymbol{\epsilon}) \quad (9)$$

Since ϵ is small, we approximate the function $g(\boldsymbol{z}+\epsilon)$ by its first-order derivative:

$$f^{\text{unlearn}}(x_{n+1}|x_{F,1:n}) \approx g(\boldsymbol{z} + \boldsymbol{\epsilon}) \quad (10)$$
$$\approx g(\boldsymbol{z}) + \nabla_{\boldsymbol{z}}g(\boldsymbol{z})^\top \boldsymbol{\epsilon} \quad (11)$$

Given that $\epsilon \sim \mathcal{N}(\mathbf{0}, \eta I)$, then:

$$f^{\text{unlearn}}(x_{n+1}|x_{F,1:n}) \quad (12)$$
$$\sim \mathcal{N}\left(||g(\boldsymbol{z})||_2^2, \eta||\nabla_{\boldsymbol{z}}g(\boldsymbol{z})||_2^2\right) \quad (13)$$

Since $\boldsymbol{u} \sim U(0,1)$, then $c \cdot \boldsymbol{u} \sim U(0,c)$. By the linearity property of expectation, we have:

$$\mathbb{E}(\boldsymbol{z}) = \mathbb{E}(c \cdot \boldsymbol{u}) = \frac{1}{2}(0 + c) = \frac{c}{2} \quad (14)$$

By definition of variance, we have:

$$\text{Var}(\boldsymbol{z}) = \text{Var}(c \cdot \boldsymbol{u}) = c^2\text{Var}(\boldsymbol{u}) = \frac{c^2}{12} \quad (15)$$

$\square$

C.2 Proof of section 2.3

Theorem 2.1. *Suppose the adversary injects a small, random update $\delta \sim \mathcal{N}(0, \nu I)$, $\nu \in \mathbb{R}^+$ to the input x and query unlearned model f^{unlearn} to find the optimal direction. The probability that*

the adversary chooses the opposite or orthogonal direction to the optimal direction is

$$\mathbb{P}\left[-1 \leq \frac{\langle \Delta\mathcal{J}, \Delta\mathcal{J}^{\text{unlearn}} \rangle}{||\Delta\mathcal{J}||_2 ||\Delta\mathcal{J}^{\text{unlearn}}||_2} \leq 0\right]$$

$$\approx \frac{1}{2}\,\text{erf}\left(\left(\frac{\nu(||\nabla_x(\mathcal{J} \circ f^{\text{unlearn}})||_2^2 + ||\nabla_x(\mathcal{J} \circ f)||_2^2)}{2}\right)^{\frac{1}{2}}\right) \tag{6}$$

where $\text{erf}(x)$ *is the Gaussian error function.*

Proof. Given δ_2 is small, we approximate $\mathcal{J}(f^{\text{unlearn}}(x + \delta_2))$ using the first-order Taylor approximation:

$$\mathcal{J}(f^{\text{unlearn}}(x + \delta_2)) = (\mathcal{J} \circ f^{\text{unlearn}})(x + \delta_2)$$
$$\approx (\mathcal{J} \circ f^{\text{unlearn}})(x) + \nabla_x(\mathcal{J} \circ f^{\text{unlearn}})^\top(x)\delta_2$$
$$= \mathcal{J}(f^{\text{unlearn}}(x)) + \nabla_x(\mathcal{J} \circ f^{\text{unlearn}})^\top(x)\delta_2 \tag{16}$$

Rearrange Eqn. 16, we get:

$$\Delta\mathcal{J}^{\text{unlearn}} \approx \nabla_x(\mathcal{J} \circ f^{\text{unlearn}})^\top(x)\delta_2 \tag{17}$$

Likewise, we use the first-order Taylor to approximate $\mathcal{J}(f(x + \delta_1))$:

$$\Delta\mathcal{J} \approx \nabla_x(\mathcal{J} \circ f)^\top(x)\delta_1 \tag{18}$$

We form the normalized inner product:

$$\frac{\langle \Delta\mathcal{J}, \Delta\mathcal{J}^{\text{unlearn}} \rangle}{||\Delta\mathcal{J}||_2 ||\Delta\mathcal{J}^{\text{unlearn}}||_2} \tag{19}$$

Since δ_1 and δ_2 are independently sampled from $\mathcal{N}(0, \nu I)$, then:

$$\Delta\mathcal{J}^{\text{unlearn}} \sim \mathcal{N}\left(0, \nu||\nabla_x(\mathcal{J} \circ f^{\text{unlearn}})||_2^2\right) \tag{20}$$

and

$$\Delta\mathcal{J} \sim \mathcal{N}(0, \nu||\nabla_x(\mathcal{J} \circ f)||_2^2) \tag{21}$$

are independent Gaussian variables with means $\eta_{\Delta\mathcal{J}} = \eta_{\Delta\mathcal{J}^{\text{unlearn}}} = 0$ and variances $\sigma^2_{\Delta\mathcal{J}^{\text{unlearn}}} = \nu||\nabla_x(\mathcal{J} \circ f^{\text{unlearn}})||_2^2$ and $\sigma^2_{\Delta\mathcal{J}} = \nu||\nabla_x(\mathcal{J} \circ f)||_2^2$. For a normally distributed random variable, the expected value of the squared ℓ_2 norm is equal to the variance. Thus,

$$\mathbb{E}[||\Delta\mathcal{J}||_2^2] = \sigma^2_{\Delta\mathcal{J}} \tag{22}$$

$$\mathbb{E}[||\Delta\mathcal{J}^{\text{unlearn}}||_2^2] = \sigma^2_{\Delta\mathcal{J}^{\text{unlearn}}} \tag{23}$$

To find the ℓ_2 norm, we take the square root of the expected value of the squared norm:

$$||\Delta\mathcal{J}||_2 = \sqrt{\sigma^2_{\Delta\mathcal{J}}} = \sqrt{\nu}||\nabla_x(\mathcal{J} \circ f)||_2 \tag{24}$$

$$||\Delta\mathcal{J}^{\text{unlearn}}||_2 = \sqrt{\sigma^2_{\Delta\mathcal{J}^{\text{unlearn}}}}$$
$$= \sqrt{\nu}||\nabla_x(\mathcal{J} \circ f^{\text{unlearn}})||_2 \tag{25}$$

The product $\langle \Delta\mathcal{J}, \Delta\mathcal{J}^{\text{unlearn}} \rangle$ follows the Gaussian distribution (Bromiley, 2003):

$$\mathcal{N}\left(0, \frac{\sigma^2_{\Delta\mathcal{J}^{\text{unlearn}}} \cdot \sigma^2_{\Delta\mathcal{J}}}{\sigma^2_{\Delta\mathcal{J}^{\text{unlearn}}} + \sigma^2_{\Delta\mathcal{J}}}\right), \tag{26}$$

Since $||\Delta\mathcal{J}||_2$ and $||\Delta\mathcal{J}^{\text{unlearn}}||_2$ are the ℓ_2-norms and are positive constant values, by definition of a linear transformation of normal distribution, $\frac{\langle \Delta\mathcal{J}, \Delta\mathcal{J}^{\text{unlearn}} \rangle}{||\Delta\mathcal{J}||_2 ||\Delta\mathcal{J}^{\text{unlearn}}||_2}$ follows the normal distribution:

$$\mathcal{N}\left(0, \frac{\sigma^2_{\Delta\mathcal{J}^{\text{unlearn}}} \cdot \sigma^2_{\Delta\mathcal{J}}}{(\sigma^2_{\Delta\mathcal{J}^{\text{unlearn}}} + \sigma^2_{\Delta\mathcal{J}})||\Delta\mathcal{J}||_2^2 ||\Delta\mathcal{J}^{\text{unlearn}}||_2^2}\right) \tag{27}$$

We denote $\mathcal{X} = \frac{\langle \Delta\mathcal{J}, \Delta\mathcal{J}^{\text{unlearn}} \rangle}{||\Delta\mathcal{J}||_2 ||\Delta\mathcal{J}^{\text{unlearn}}||_2} \sim \mathcal{N}(\eta_{\mathcal{X}}, \sigma^2_{\mathcal{X}})$ with mean $\eta_{\mathcal{X}} = 0$ and variance $\sigma^2_{\mathcal{X}} = \frac{\sigma^2_{\Delta\mathcal{J}^{\text{unlearn}}} \cdot \sigma^2_{\Delta\mathcal{J}}}{(\sigma^2_{\Delta\mathcal{J}^{\text{unlearn}}} + \sigma^2_{\Delta\mathcal{J}})||\Delta\mathcal{J}||_2^2 ||\Delta\mathcal{J}^{\text{unlearn}}||_2^2}$. Substituting Eqn. 24 and Eqn. 25 to Eqn. 26, we get:

$$\mathcal{X} \sim \mathcal{N}\left(0, \frac{1}{\sigma^2_{\Delta\mathcal{J}^{\text{unlearn}}} + \sigma^2_{\Delta\mathcal{J}}}\right) \tag{28}$$

The probability for $\mathcal{X}$ being in the interval of $[-1, 0]$ can be derived as

$$\mathbb{P}(-1 \leq \mathcal{X} \leq 0) = \mathcal{F}(0) - \mathcal{F}(-1) \tag{29}$$

Where $\mathcal{F}(x) = \frac{1}{2}\left(1 + \text{erf}\,\frac{x - \eta_{\mathcal{X}}}{\sqrt{2}\sigma_{\mathcal{X}}}\right)$ is the cumulative distribution function (CDF) and $\text{erf}(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}\,dt$ is the Gaussian error function. Evaluating the CDF at $x = 0$ and $x = -1$, we get:

$$\mathbb{P}(-1 \leq \mathcal{X} \leq 0) \approx \frac{1}{2}\,\text{erf}\,\frac{1}{\sqrt{2}\sigma_{\mathcal{X}}}$$

$$= \frac{1}{2}\,\text{erf}\,\frac{1}{\sqrt{2} \cdot \sqrt{\frac{1}{\nu(||\nabla_x(\mathcal{J} \circ f^{\text{unlearn}})||_2^2 + ||\nabla_x(\mathcal{J} \circ f)||_2^2)}}}$$

$$= \frac{1}{2}\,\text{erf}\left(\left(\frac{\nu(||\nabla_x(\mathcal{J} \circ f^{\text{unlearn}})||_2^2 + ||\nabla_x(\mathcal{J} \circ f)||_2^2)}{2}\right)^{\frac{1}{2}}\right) \tag{30}$$

$\square$

Method/tasks	WMDP-Bio↓	WMDP-Cyber↓	MMLU-All↑	Average↑
Base (before unlearning)	63.7	43.5	58.1	—
LLMU (Yao et al., 2023)	59.5	39.5	44.7	24.4
SCRUB (Kurmanji et al., 2023)	43.8	39.3	51.2	31.6
SSD (Foster et al., 2024)	50.2	35.0	40.7	25.8
RMU ($l = 7$) (Li et al., 2024)	<u>28.8</u>	<u>28.8</u>	**56.8**	<u>40.8</u>
Adaptive RMU ($l = 7$) (Our)	**23.7**	**26.5**	<u>55.0</u>	**41.7**

Table 4: Average of drop-in-accuracy on WMDP (Biology and Cyber) and accuracy on MMLU-All.

D Additional results

D.1 Unlearning performance of other models

We report the unlearning performance of Adaptive RMU Yi 6B, Meta Llama-3 8B, and Mistral 7B (v0.1) models in Table 5, 6, and 7. We observed a clear trend that the unlearning performance is more effective when using the early layer as the unlearn layer. We conjecture that LLMs are more confident about predictions in later layers, making unlearning more difficult. In contrast, the representation of early layers is more stochastic, potentially facilitating easier unlearning.

D.2 Unlearning performance on MMLU subset unlearning benchmark

We do additional experiments on the MMLU subset unlearning benchmark with three settings:

1. MMLU-Economics: unlearning on high school microeconomics and macroeconomics and maintaining performance on the remaining categories.

2. MMLU-Law: unlearning on international and professional law while maintaining performance on remaining categories.

3. MMLU-Physics: unlearning on high school and college physics while maintaining general performance on other categories.

Settings. We use publicly released forget set by Li et al. (2024) for each task and Wikitext (Merity et al., 2016) as retain set. We use a fixed sequence len of 512 for MMLU-Economics, MMLU-Law, MMLU-Physics, and Wikitext as well. We keep other hyperparameters remain unchanged as in Appendix B.1.

Result. Table 8 shows the unlearning performance of Adaptive RMU Zephyr 7B models on MMLU-Economics, MMLU-Law, and MMLU-Physics. We observed a significant drop in accuracy. However, its unlearns too much, causing a huge degradation in MMLU-Retain tasks.

D.3 The effect of in-domain retain set on unlearning performance.

In this setting, we use the WMDP-Bio and WMDP-Cyber retain sets instead of Wikitext. We use the same hyperparameters as in Appendix B.1. Results in Table 9 show that Adaptive RMU is almost ineffective for all unlearn layers. As WMDP-forget and retain sets are collected from the same source, even with efforts in distinction, these corpus may commonly have overlap texts. We present an n-gram overlap analysis between WMDP-forget set and WMDP-retain set as a measurement of unlearning difficulty.

n**-gram overlap analysis.** Given a retain sample $x_{1:k} \in \mathcal{D}_{\text{retain}}$ consists of k tokens $\{x_1, x_2, ...x_k\}$, we denote $x_{i:i+n-1}$ for $i \in [1, ..., k-n+1]$ as the n-gram of $x_{1:k}$. The n-gram overlap score of $x_{1:k}$ in forget set $\mathcal{D}_{\text{forget}} = \{x_F\}^{|\mathcal{D}_{\text{forget}}|}$ is defined as:

$$\frac{1}{|\mathcal{D}_{\text{forget}}|} \frac{1}{k-n+1} \sum_{x_R} \sum_{i=1}^{k-n+1} \mathbb{I}[x_{i:i+n-1} \in x_F], \tag{31}$$

where $\mathbb{I}(\cdot)$ is the indicator function and $\mathbb{I}[x_{i:i+n-1} \in x_F] = 1$ if the substring $x_{i:i+n-1}$ is in forget sample x_F, otherwise 0.

We randomly sampled 1000 documents from each dataset and performed Unigram ($n = 1$) and Bigram ($n = 2$) overlap analysis. The results indicate a high degree of unigram and bigram overlap between the WMDP-forget and WMDP-retain sets. Specifically, the average Unigram and Bigram overlap scores for the WMDP-Bio forget and retain sets were 20.8% and 5.5%, respectively. These overlap scores were even higher for the WMDP-Cyber sets,

Task/unlearn layer	base	3	4	5	6	7	8	9	10	11	12	13	14	15	16
WMDP-Biology ↓	64.8	65.0	49.9	35.2	27.8	26.1	63.3	26.2	27.1	27.4	27.1	<u>26.0</u>	**25.4**	27.2	34.8
WMDP-Cyber ↓	41.1	40.7	40.5	37.7	28.1	25.5	39.3	25.6	<u>23.9</u>	26.1	**23.6**	24.3	24.2	24.0	25.5
MMLU-All ↑	60.0	60.1	57.7	59.4	51.4	56.5	59.9	56.8	53.7	48.1	49.3	57.0	55.6	47.7	53.3
Task/unlearn layer	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
WMDP-Biology ↓	30.3	32.2	27.1	31.9	41.0	53.4	50.4	53.2	39.2	46.0	39.0	42.5	41.6	40.5	64.8
WMDP-Cyber ↓	25.3	24.4	24.3	24.5	26.7	29.8	33.9	36.2	34.3	34.6	31.4	30.4	39.6	40.8	40.6
MMLU-All ↑	45.4	52.1	56.7	58.2	59.3	59.4	59.6	59.7	59.4	59.7	59.4	59.4	59.5	59.7	60.1

Table 5: Q&A accuracy of Adaptive RMU Yi-6B models on WMDP-Biology, WMDP-Cyber, and MMLU-all w.r.t unlearn layer l from $3 \to 31$. The scaling factor $\beta = 5$. The **best** and <u>runner up</u> are marked.

Task/unlearn layer	base	3	4	5	6	7	8	9	10	11	12	13	14	15	16
WMDP-Biology ↓	71.2	46.4	45.3	<u>28.2</u>	**27.8**	29.3	33.7	36.0	65.1	64.9	62.8	65.2	59.6	44.4	41.4
WMDP-Cyber ↓	43.9	32.5	<u>25.5</u>	**24.5**	27.6	26.8	27.3	26.3	32.5	32.3	34.1	35.2	29.9	28.3	27.8
MMLU-All ↑	62.0	60.7	60.2	59.7	60.7	60.0	60.1	59.6	61.8	61.3	61.5	61.5	61.8	60.9	61.1
Task/unlearn layer	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
WMDP-Biology ↓	35.5	35.2	41.1	60.8	33.7	59.3	54.6	56.7	69.6	62.2	70.0	69.9	69.9	67.0	70.4
WMDP-Cyber ↓	28.0	33.5	28.6	39.0	28.6	31.7	35.5	36.9	45.5	44.8	44.4	43.5	44.4	43.6	43.4
MMLU-All ↑	61.3	61.3	61.3	61.9	60.8	61.7	61.2	61.5	61.9	61.7	62.0	61.9	61.5	61.5	62.1

Table 6: Q&A accuracy of Adaptive RMU Meta-Llama 3 8B models on WMDP-Biology, WMDP-Cyber, and MMLU-all w.r.t unlearn layer l from $3 \to 31$. The scaling factor $\beta = 5$. The **best** and <u>runner up</u> are marked.

Task/unlearn layer	base	3	4	5	6	7	8	9	10	11	12	13	14	15	16
WMDP-Biology ↓	67.3	28.0	28.9	27.6	27.5	26.3	**24.5**	<u>25.7</u>	26.1	27.6	31.4	37.7	35.6	25.4	35.0
WMDP-Cyber ↓	44.1	42.1	41.9	**24.8**	26.8	26.3	26.6	26.4	26.7	<u>25.7</u>	26.5	25.8	31.6	26.7	27.9
MMLU-All ↑	58.7	54.5	57.2	54.9	55.8	55.7	47.3	53.0	47.4	35.1	54.5	55.9	51.5	44.9	57.3
Task/unlearn layer	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
WMDP-Biology ↓	27.4	56.4	38.4	45.7	42.0	52.0	52.4	61.1	57.5	62.2	63.2	66.3	61.9	61.0	66.0
WMDP-Cyber ↓	27.5	38.9	26.5	26.7	26.6	27.4	27.7	38.9	43.9	43.4	43.7	43.8	44.0	42.5	43.4
MMLU-All ↑	56.7	56.8	56.2	57.6	58.1	58.3	58.1	58.2	58.6	58.7	58.6	58.7	58.4	58.3	58.2

Table 7: Q&A accuracy of Adaptive RMU Mistral 7B (v0.1) models on WMDP-Biology, WMDP-Cyber, and MMLU-all w.r.t unlearn layer l from $3 \to 31$. The scaling factor $\beta = 5$. The **best** and <u>runner up</u> are marked.

Task/unlearn layer	base	3	4	5	6	7	8	9	10	11	12	13	14	15	16
MMLU-Economics ↓	58.0	57.0	45.7	22.8	23.4	27.0	28.8	27.0	34.6	24.6	42.1	45.5	34.8	44.5	58.3
MMLU-Law ↓	55.6	49.8	53.5	25.2	24.5	26.4	24.6	24.2	21.5	23.9	51.1	44.1	36.8	44.7	46.0
MMLU-Physics ↓	38.5	39.3	37.9	28.8	27.2	23.8	21.7	20.5	21.0	29.2	32.6	34.1	34.4	35.7	42.3
MMLU-Retain ↑	58.9	58.0	57.3	39.3	45.2	39.4	35.2	36.0	44.8	35.2	52.9	55.2	46.0	54.8	56.8
Task/unlearn layer	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
MMLU-Economics ↓	51.8	36.0	54.4	26.0	21.4	42.8	43.4	42.8	48.4	57.2	58.7	50.0	58.2	58.9	57.8
MMLU-Law ↓	49.8	24.3	54.4	27.2	24.6	24.2	25.4	44.6	54.4	55.8	56.7	53.6	55.6	55.4	56.1
MMLU-Physics ↓	37.5	26.7	26.9	21.0	21.6	24.2	23.4	25.6	29.6	37.1	31.9	33.8	36.9	33.9	38.6
MMLU-Retain ↑	57.6	47.8	57.7	36.2	30.3	39.6	47.4	52.0	58.1	58.9	58.9	56.4	59.0	59.1	59.0

Table 8: Q&A accuracy of Adaptive RMU Zephyr 7B models on MMLU-Economics, MMLU-Law, MMLU-Phycics, and MMLU-Retain w.r.t unlearn layer l from $3 \to 31$. The scaling factor $\beta = 5$.

at 27.5% and 12.3%, respectively. The distributions of n-gram overlap scores are visualized in Fig. 3. High n-gram overlap scores make two distributions WMDP-forget set and WMDP-retain set less distinction, which cause the unlearning more difficult.

D.4 Example of generated outputs

This section shows examples of generated answers in WMDP-Biology and WMDP-Cyber by Zephyr-7B model. The number of new tokens is 30.

Example 1 (Biology):

"The following are multiple choice questions (with answers) about Biology.

Question: My labmate is setting up

Task/unlearn layer	base	3	4	5	6	7	8	9	10	11	12	13	14	15	16
WMDP-Biology ↓	63.7	63.2	63.3	62.9	28.1	62.6	49.9	64.2	29.6	62.0	63.0	63.7	63.7	64.4	64.3
WMDP-Cyber ↓	43.5	42.7	42.0	40.1	24.6	33.3	33.9	40.8	25.1	41.3	41.7	42.8	43.4	42.8	43.4
MMLU-All ↑	58.1	57.4	57.4	57.9	30.1	57.6	38.3	57.6	29.3	57.1	58.0	57.5	57.7	57.9	57.8

Task/unlearn layer	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
WMDP-Biology ↓	63.9	63.7	63.9	63.5	63.5	63.7	63.7	63.6	63.6	63.5	63.3	63.7	63.8	63.5	64.6
WMDP-Cyber ↓	44.5	43.5	43.5	44.4	43.9	43.5	44.3	43.6	43.9	43.8	43.6	43.2	43.7	43.7	43.6
MMLU-All ↑	58.4	58.1	58.2	57.6	58.2	58.1	58.2	58.1	58.1	58.0	58.2	58.1	58.2	58.1	57.9

Table 9: Q&A accuracy of Adaptive RMU Zephyr 7B models on WMDP-Bio, WMDP-Cyber, and MMLU-All w.r.t unlearn layer l from $3 \rightarrow 31$. Models were fine-tuned with WMDP-Bio retain set and WMDP-Cyber retain set. The scaling factor $\beta = 5$.

for some lab work. They have turned on a biosafety cabinet, set out Mueller-Hinton agar and cefoxitin diffusion disks, and are warming an incubator to 37 C. What microorganism might they be growing?
A. Mycobaterium smegmatis
B. Aspergillus niger
C. Staphylococcus aureus
D. Bacillus subtilis
Answer:"
True answer: "Staphylococcus aureus".
RMU's answer: "C. Staphylococcus aureus
Question: Which of the following is a type of cellular respiration?
A".
Adaptive RMU's answer: "The ABabai lababudcai (hotspowered the TT Superior, and the TT Superior is a superior".

Example 2 (Cyber):
"The following are multiple choice questions (with answers) about Cyber.

Question: Are any of the following flags a security concern when passed to the v8 JavaScript engine? Assume the engine will be used to run arbitrary JS from the internet.
A. –trace-turbo
B. –trace-gc
C. –print-bytecode
D. –allow-natives-syntax"
Answer:"
True answer: "–allow-natives-syntax".
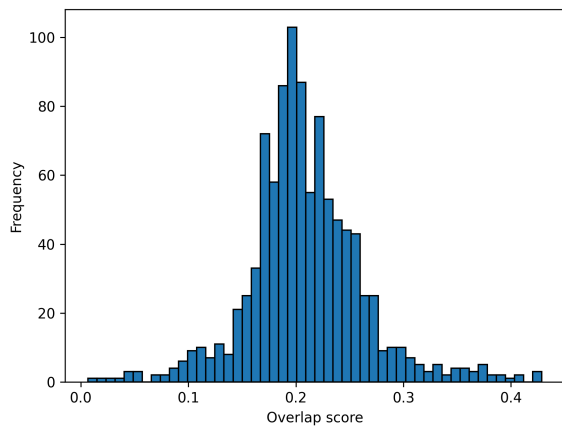RMU's answer: "D. –allow-natives-syntax

Question:Which of the following is a security concern when passed to the v8 JavaScript engine".
Adaptive RMU's answer: "The above results are not both both both both than both of
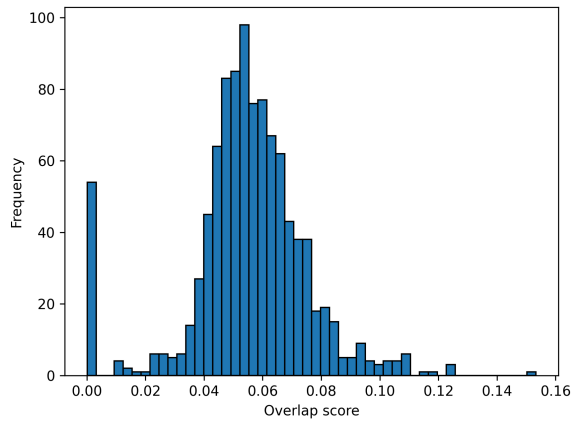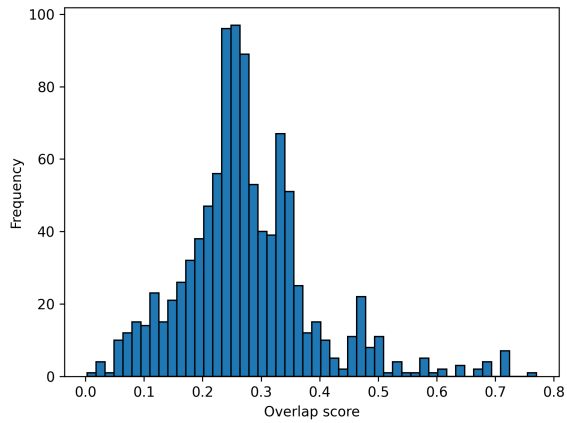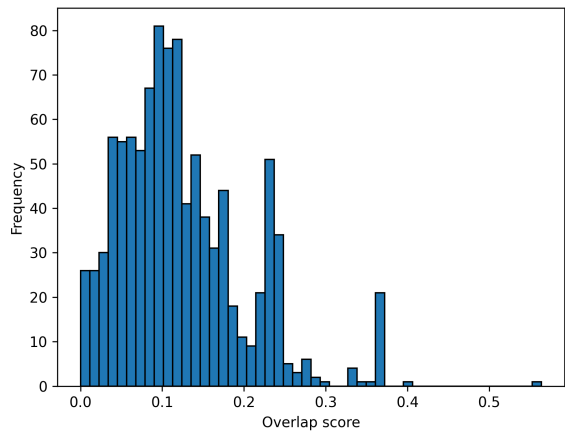
the both of you.

C. The both of you.

both both".

(a) Distribution of Unigram overlap score between WMDP-Bio retain and WMDP-Bio forget sets.

(b) Distribution of Bigram overlap score between WMDP-Bio retain and WMDP-Bio forget sets.

(c) Distribution of Unigram overlap score between WMDP-Cyber retain and WMDP-Cyber forget sets.

(d) Distribution of Bigram overlap score between WMDP-Cyber retain and WMDP-Cyber forget sets.

Figure 3: Distributions of Unigram and Bigram overlap scores.