Towards Reliable Evaluations of Machine Unlearning

AAAI 2026 Tutorial

Content

Setting

- Practical Setting (Arxiv:2406.08288)

Construct machine unlearning scenarios with decoupled label domains and target concepts.

Metric

- Reliable Metrics (ICLR'25)

Focus on the optimization of evaluation metrics for machine unlearning, and clarify the most reliable metrics.

- LLM Judgement (Arxiv:2510.19422)

Focus on addressing the cumulative decline and cascading degradation in continual unlearning scenarios.

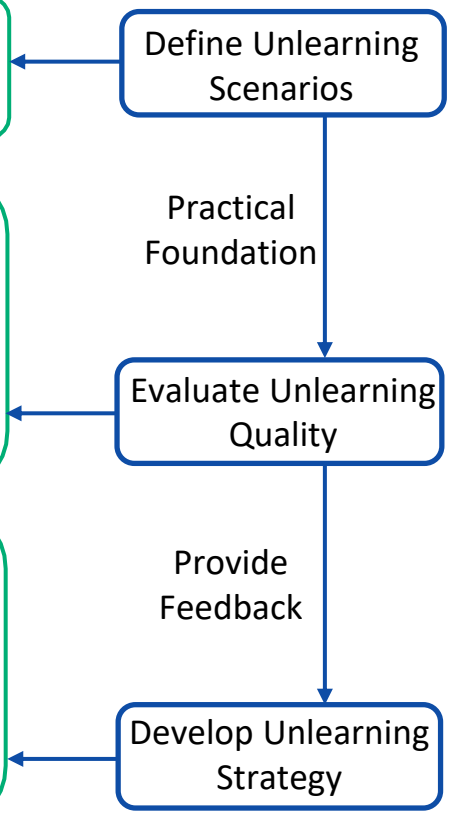
Method

- Gradient Analysis (ICLR'25)

Analyze the mechanism and optimization direction of the unlearning function from the perspective of gradients.

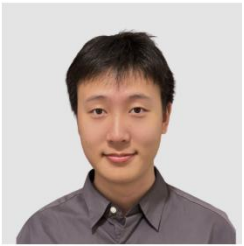
- Model Patching Analysis (ICML'25 Workshop)

Balance the forget quality and model utility based on layer-wise fragility estimation.



Decoupling the Class Label and the Target Concept in Machine Unlearning

Jianing Zhu, Bo Han, Jiangchao Yao, Jianliang Xu, Gang Niu,
Masashi Sugiyama

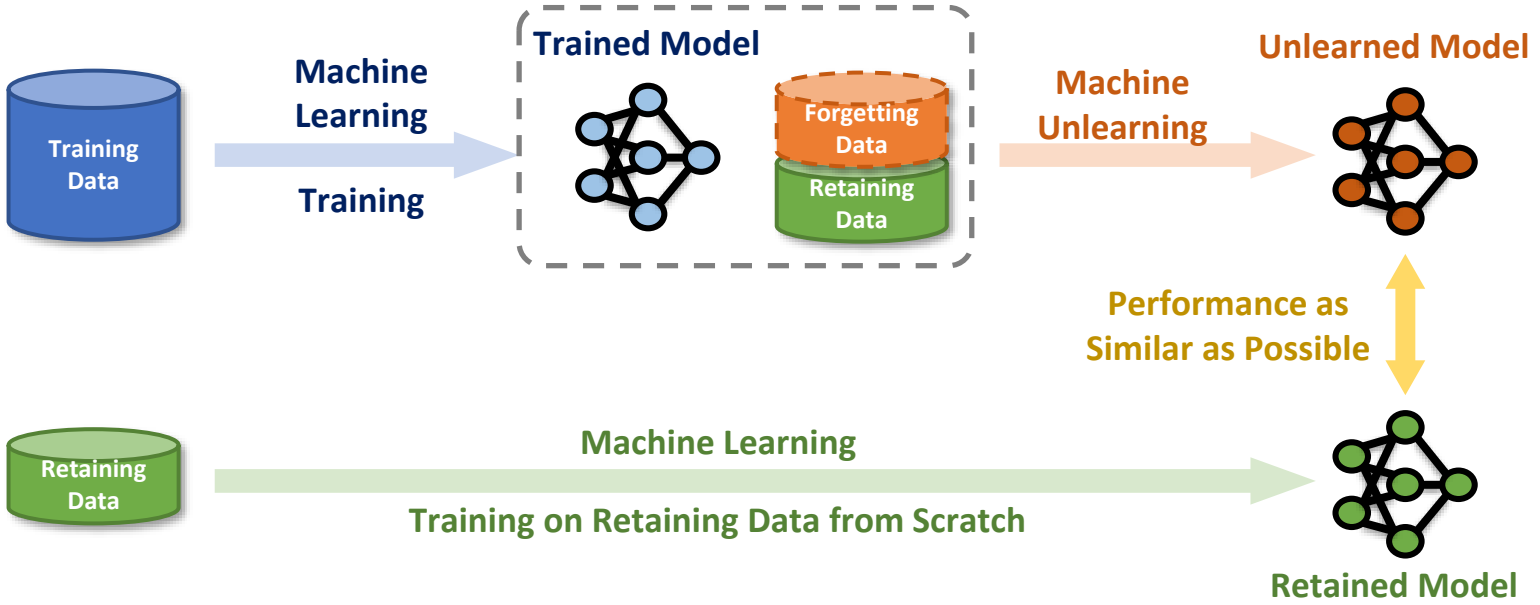


Dr. Jianing Zhu

Dr. Jiangchao Yao

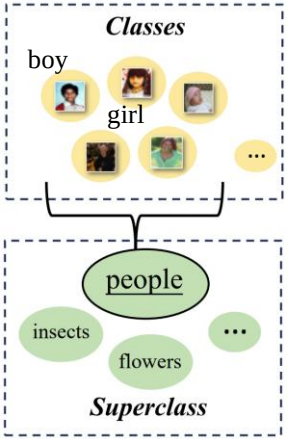
Background | Machine Unlearning

- ✓ **Machine unlearning** aims to remove the influence of the **forgetting data** from a trained model, such that it behaves similarly to a model (termed Retrained) retrained from scratch on the **retaining data**.



Decoupling the Class Label and the Target Concept in Machine Unlearning. *Arxiv:2406.08288.*

Background | Label Domain Mismatch

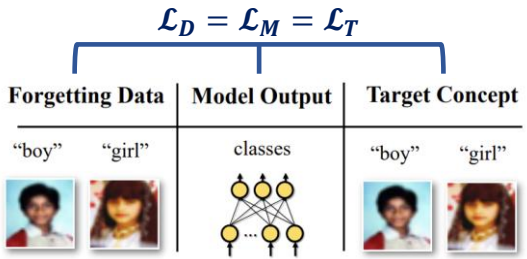


Label Domain of CIFAR-100

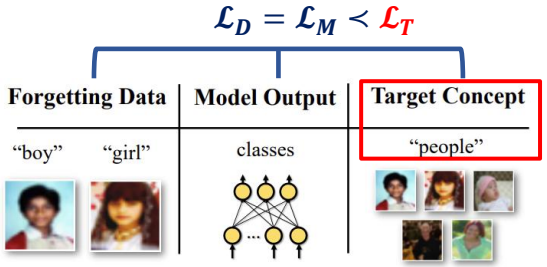
Label domains:

- ✓ $\mathcal{L}_D$: label domain of forgetting data.
- ✓ $\mathcal{L}_M$: label domain of the model output.
- ✓ $\mathcal{L}_T$: label domain of unlearning target concept.

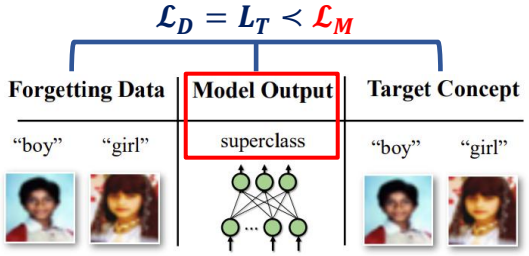
Four Types of Unlearning Scenarios



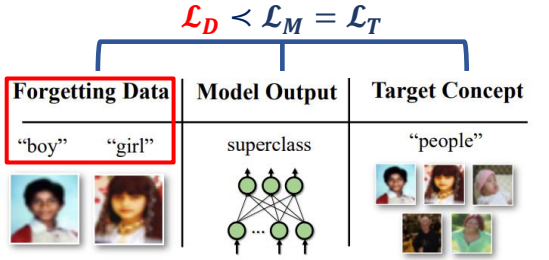
(a) **All Matched**
(Conventional Unlearning)



(b) Called **Target Mismatch**



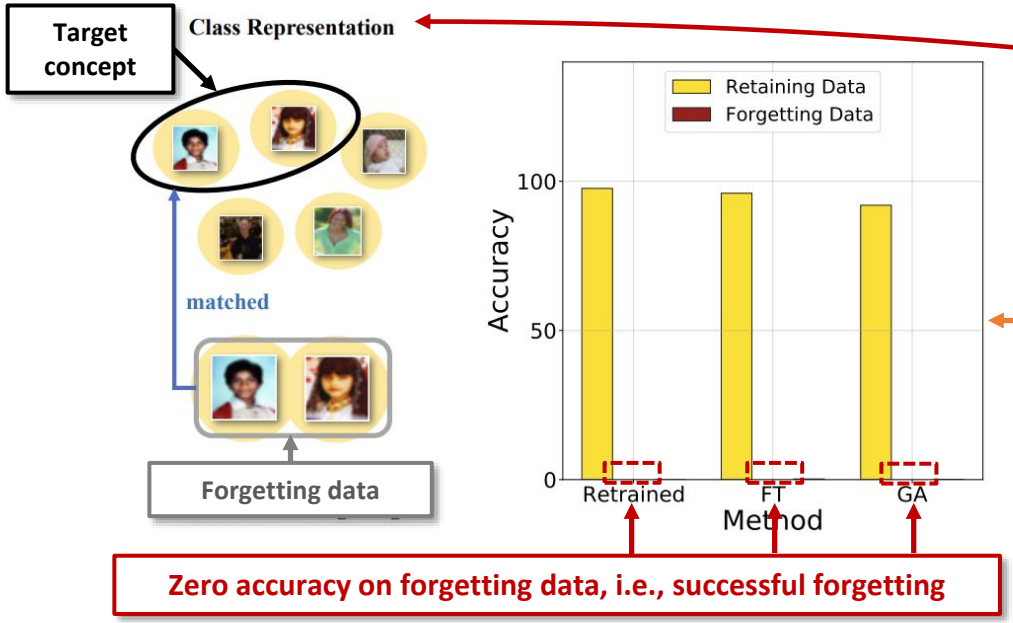
(c) Called **Model Mismatch**



(d) Called **Data Mismatch**

Decoupling the Class Label and the Target Concept in Machine Unlearning. *Arxiv:2406.08288.*

Conventional Scenario | All Matched Forgetting



(a) **All Matched**
(Conventional Unlearning)

Observation 1: Representations are consistent.

Observation 2: FT and **GA** can achieve **similar performance** on retraining and forgetting data like **Retrained**.

Zero accuracy on forgetting data, i.e., successful forgetting

Note:
- ✓ **Retrained:** Retrain Model on Retaining Data.
- ✓ **FT:** Fine-tuning (Unlearning Method).
- ✓ **GA:** Gradient Ascent (Unlearning Method).

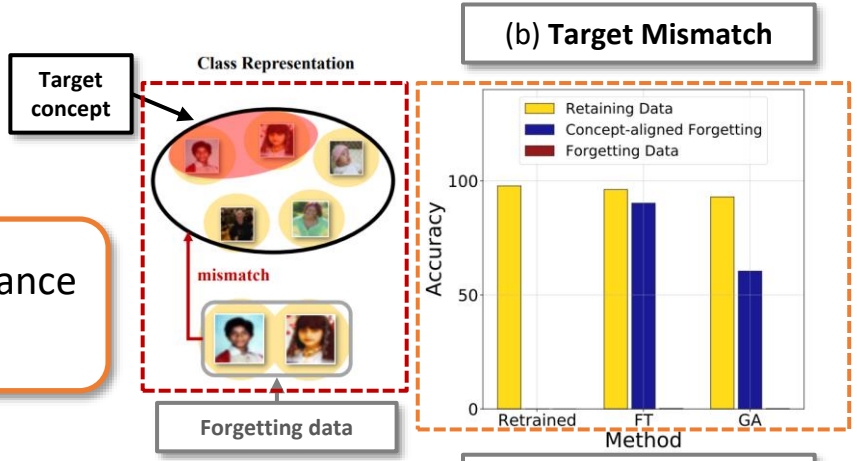
This is exactly what we aim for.

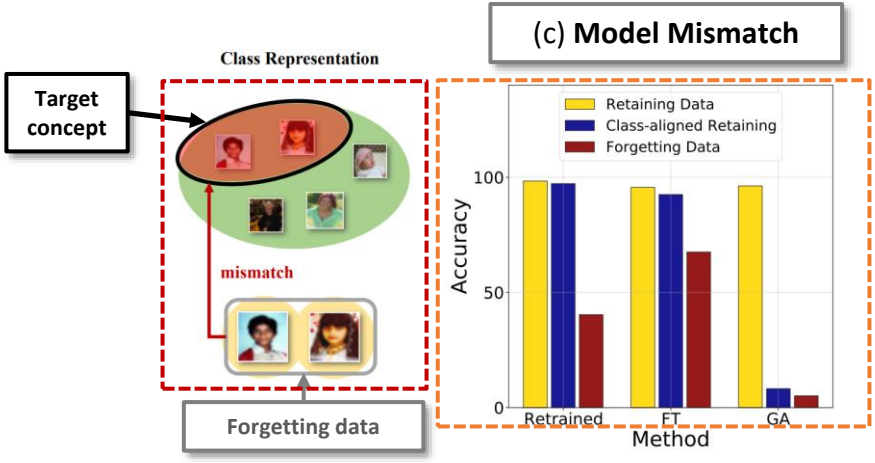
Decoupling the Class Label and the Target Concept in Machine Unlearning. *Arxiv:2406.08288.*

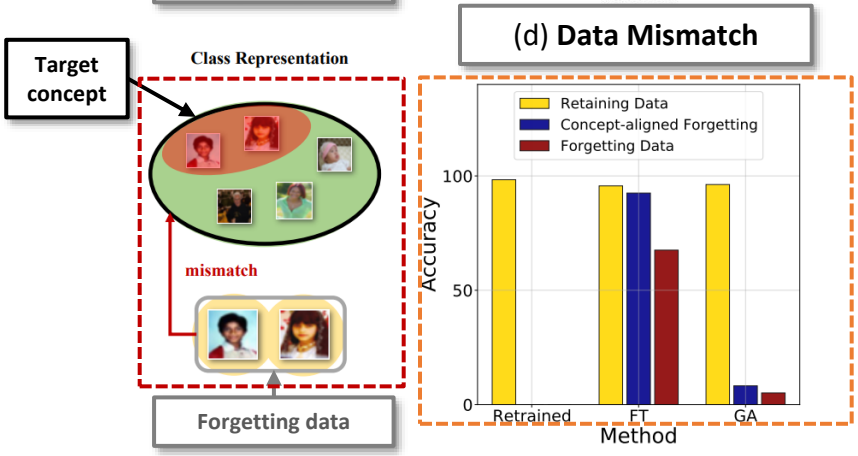
Challenge | Three Types of Mismatched Scenarios

Observation 3: Class representation mismatch issue.

Observation 4: FT and **GA** show different performance gaps compared with the **Retrained** models.





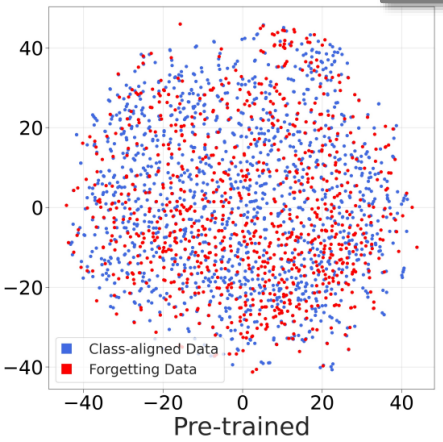


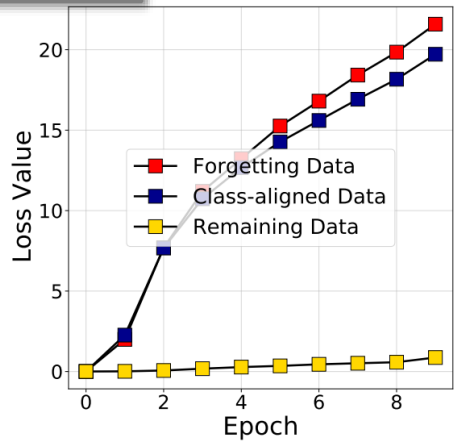
Decoupling the Class Label and the Target Concept in Machine Unlearning. *Arxiv:2406.08288.*

Observation | Representation Entangled

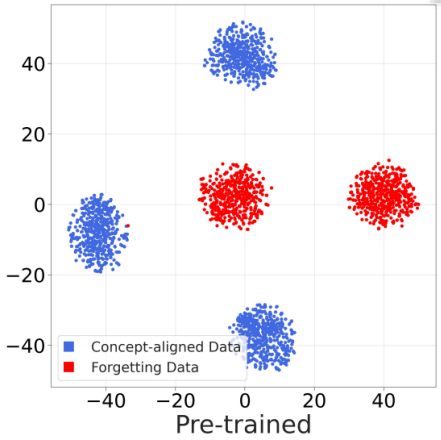
- **Visualization of the learned features** from the model trained by (left) superclass and (right) classes.
- **Loss value of** forgetting data, concept/class-aligned data, and the remaining data during GA.

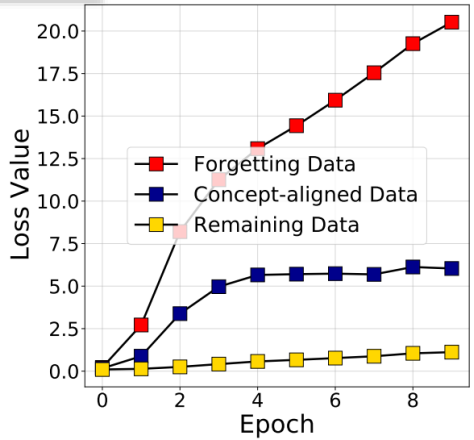
Superclass





Class



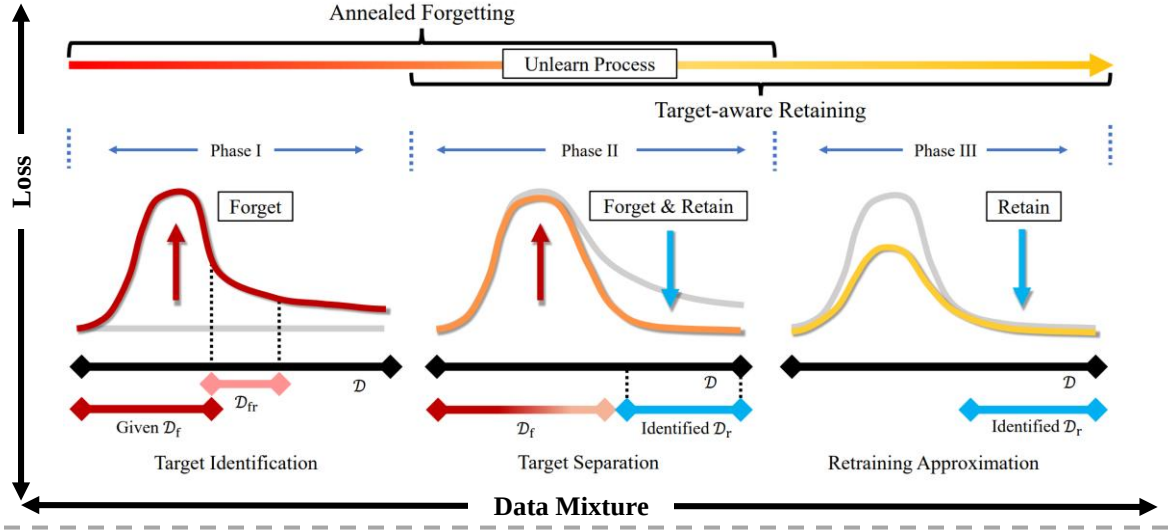


Observation 5: Representations of forgetting data and affected retaining data are closely entangled.

Observation 6: Unlearning of the forgetting data can unavoidably affect the representation of the other part.

Decoupling the Class Label and the Target Concept in Machine Unlearning. *Arxiv:2406.08288.*

Methodology | Overview of TARF



TARget-aware Forgetting (TARF)

- ✓ Two terms consist of **Annealed Forgetting** and **Target-aware Retaining**.
- ✓ The training dynamics go through **Three Phases.**

Learning Objective of TARF

Phase II: Target Separation

Phase I: Target Identification | **Phase III: Retraining Approximation**

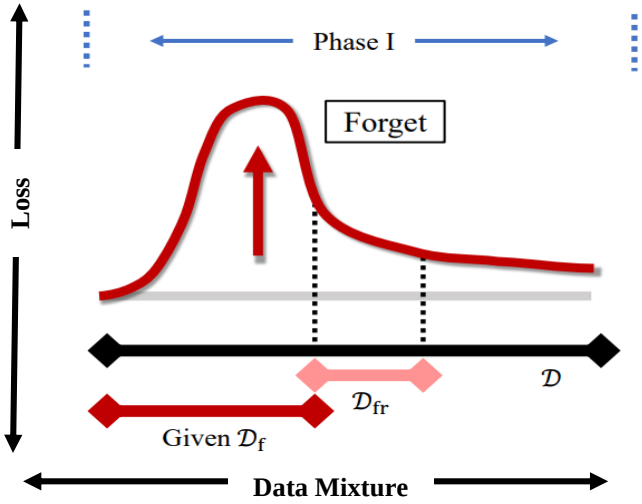
$$L_{\text{TARF}} = \boldsymbol{k(t)} \cdot \left(-\frac{1}{|\mathcal{D}_{\text{f}}|} \sum_{(x,y)\sim\mathcal{D}_{\text{f}}} \ell(f(x), y) \right) + \frac{1}{|\mathcal{D}_{\text{un}}|} \sum_{(x,y)\sim\mathcal{D}_{\text{un}}} \ell(f(x), y) \cdot \boldsymbol{\tau(x, y, t)}$$

Annealed Forgetting | **Target-aware Retaining**

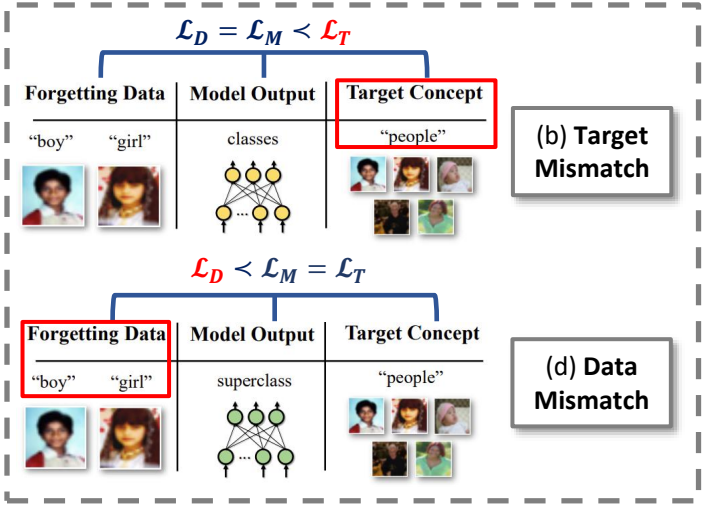
Training phases are controlled by t.

Decoupling the Class Label and the Target Concept in Machine Unlearning. *Arxiv:2406.08288.*

Methodology | Phase I: Target Identification



Aim to Address



- ✓ $\mathcal{D}_{\text{f}}$: Forgetting Data.
- ✓ $\mathcal{D}_{\text{fr}}$: False Retaining Data.

Objective of TARF-Phase-I

Phase I: Target Identification

$$L_{\text{TARF-Phase-I}} = \boldsymbol{k(t)} \cdot \left(-\frac{1}{|\mathcal{D}_{\text{f}}|} \sum_{(x,y)\sim\mathcal{D}_{\text{f}}} \ell(f(x), y) \right)$$

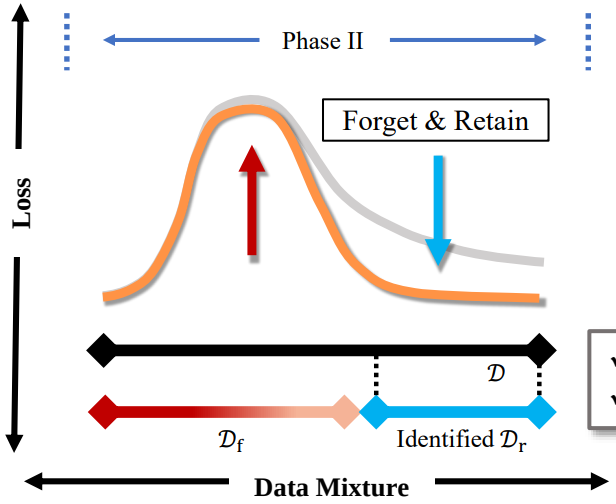
Annealed Forgetting

Goal of TARF Phase I:

- ✓ Learn the **representations** of forgetting data.
- ✓ Identify **potential forgetting data**, i.e., false retaining data, from remaining data.

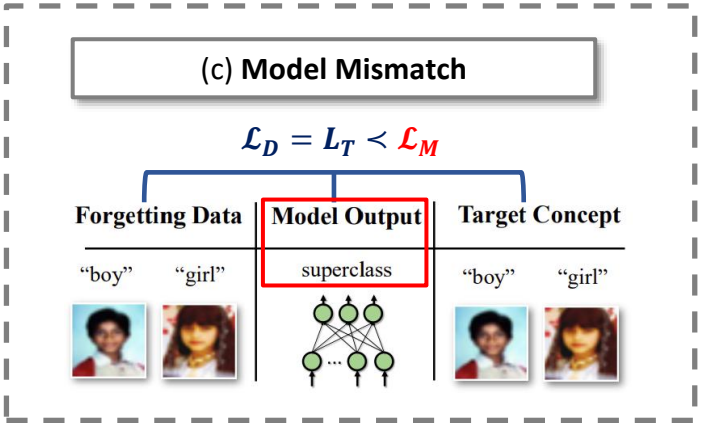
Decoupling the Class Label and the Target Concept in Machine Unlearning. *Arxiv:2406.08288.*

Methodology | Phase II: Target Separation



Aim to Address

- ✓ $\mathcal{D}_f$: Forgetting Data.
- ✓ $\mathcal{D}_r$: Retaining Data.



Objective of TARF-Phase-II

Phase II: Target Separation

$$L_{\text{TARF-Phase-II}} = k(t) \cdot \left(-\frac{1}{|\mathcal{D}_f|} \sum_{(x,y)\sim\mathcal{D}_f} \ell(f(x), y) \right) + \frac{1}{|\mathcal{D}_{un}|} \sum_{(x,y)\sim\mathcal{D}_{un}} \ell(f(x), y) \cdot \tau(x, y, t)$$

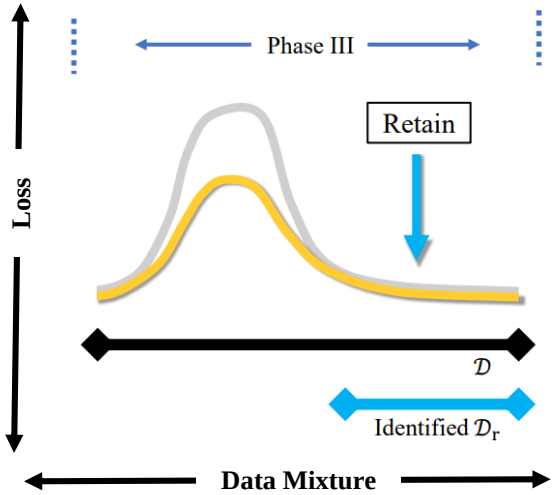
Annealed Forgetting | **Target-aware Retaining**

Goal of TARF Phase II:

- ✓ Learn the **representations** of forgetting data and retaining data.
- ✓ Encourage the model to **deconstruct the target concept** and **reconstruct representations** of the retaining part.

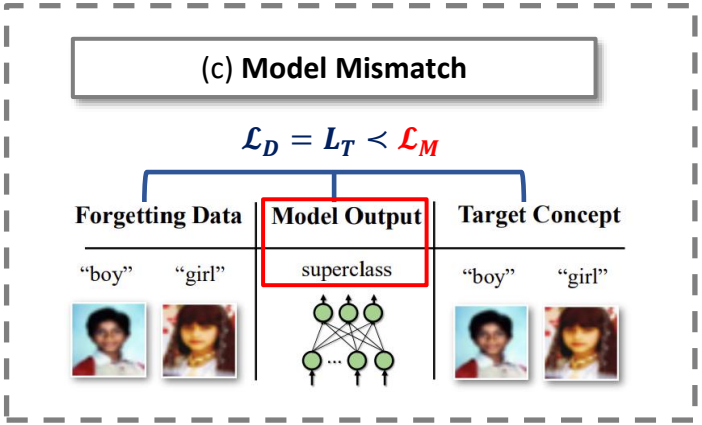
Decoupling the Class Label and the Target Concept in Machine Unlearning. *Arxiv:2406.08288.*

Methodology | Phase III: Retraining Approximation



Aim to Address

- ✓ $\mathcal{D}_r$: Retaining Data.
- ✓ $\mathcal{D}_{un}$: Remaining Data, including True Retaining Data and False Retaining Data.



Objective of TARF-Phase-III

Phase III: Retraining Approximation

$$L_{\text{TARF-Phase-III}} = \frac{1}{|\mathcal{D}_{\text{un}}|} \sum_{(x,y)\sim\mathcal{D}_{\text{un}}} \ell(f(x), y) \cdot \tau(x, y, t)$$

Target-aware Retaining

Goal of TARF Phase III:

- ✓ Learn to tune the **representations** of retaining data.
- ✓ **Prevent excessive forgetting**.
- ✓ **Approximate the retraining objective**.

Decoupling the Class Label and the Target Concept in Machine Unlearning. *Arxiv:2406.08288.*

Experiments | Empirical Evaluations

Type / $\mathcal{D}$	Dataset	CIFAR-10						CIFAR-100					
	Method / Metrics	UA	RA	TA	MIA	Gap↓	TIME↓	UA	RA	TA	MIA	Gap↓	TIME↓
All matched	Retrained (Ref.)	0.00	99.51	94.69	100.00	-	43.3	0.00	97.85	76.03	100.00	-	43.2
	FT [58]	1.07	98.62	92.36	100.00	1.07	4.43	0.67	96.32	72.34	100.00	1.47	5.02
	RL [56]	4.13	97.65	91.23	100.00	2.36	4.88	1.00	96.09	72.00	100.00	1.70	4.96
	GA [28]	0.49	95.24	88.17	99.78	2.88	**0.25**	1.33	94.74	68.56	99.89	3.01	**0.06**
	IU [29]	0.22	88.15	82.38	99.96	5.99	0.45	0.00	37.61	29.58	100.00	26.67	0.51
	BS [6]	25.04	87.94	80.90	88.67	15.43	0.82	4.60	90.18	63.66	99.55	6.27	0.78
	L_1-sparse [30]	0.00	94.20	89.77	100.00	2.56	4.39	0.00	94.60	71.57	100.00	1.93	4.39
	SalUn [11]	0.00	91.32	86.87	100.00	4.00	5.65	0.00	75.34	62.14	100.00	9.10	5.75
	SCRUB [37]	0.00	99.94	91.00	100.00	1.03	2.88	0.00	99.98	76.75	100.00	**0.71**	3.23
	TARF (ours)	0.00	98.23	91.95	100.00	**1.01**	4.21	0.00	96.90	72.53	100.00	1.11	4.68
Model mismatch	Retrained (Ref.)	87.76	99.58	95.91	20.57	-	43.8	88.22	98.58	78.50	25.78	-	43.8
	FT [58]	94.67	98.53	93.56	9.56	5.33	4.29	92.67	95.02	79.34	16.33	4.58	4.86
	RL [56]	53.69	97.85	92.39	96.60	28.84	4.82	80.11	95.83	79.83	99.00	21.35	4.93
	GA [28]	5.76	86.99	82.20	94.98	45.68	**0.25**	6.78	94.83	76.96	97.78	39.68	**0.06**
	IU [29]	23.69	87.34	82.57	89.87	39.74	0.44	34.67	96.83	79.08	86.44	29.14	0.49
	BS [6]	10.29	50.77	49.39	95.96	62.05	0.79	18.11	95.90	72.28	95.22	37.14	0.89
	L_1-sparse [30]	93.11	94.76	91.63	14.44	5.15	4.24	90.22	94.78	78.81	18.88	3.25	5.00
	SalUn [11]	8.91	93.95	84.38	99.32	43.69	6.04	66.33	78.83	70.78	77.00	25.15	5.97
	SCRUB [37]	95.14	99.81	94.22	15.38	3.61	3.06	91.44	99.74	79.23	21.11	2.45	4.12
	TARF (ours)	91.11	97.49	92.49	17.82	**2.90**	4.31	86.67	97.05	80.07	26.00	**1.21**	4.81
Target mismatch	Retrained (Ref.)	0.00	99.38	93.85	100.00	-	52.1	0.00	97.85	73.72	100.00	-	53.2
	FT [58]	50.43	98.47	91.65	50.44	25.78	4.38	58.18	96.32	72.53	46.76	28.54	5.00
	RL [56]	51.25	97.56	90.90	56.23	24.95	4.79	58.89	96.05	72.20	46.98	28.81	4.93
	GA [28]	40.82	97.01	89.51	64.32	20.80	**0.26**	21.38	96.64	70.22	90.67	8.86	**0.05**
	IU [29]	44.51	88.07	81.80	58.73	27.29	0.44	30.62	37.19	29.58	63.69	42.93	0.50
	BS [6]	53.62	88.65	75.39	76.33	26.62	0.82	40.44	98.32	68.66	85.16	15.20	0.97
	L_1-sparse [30]	49.47	93.61	88.83	51.24	27.26	4.38	56.09	94.63	72.00	48.04	28.25	4.78
	SalUn [11]	46.63	91.08	86.31	60.94	25.38	5.90	59.64	75.52	62.37	65.96	27.35	5.81
	SCRUB [37]	49.98	99.94	92.10	50.18	25.53	2.89	59.64	99.99	75.32	44.89	29.90	3.52
	TARF (ours)	0.06	97.57	90.81	100.00	**1.23**	4.23	0.31	97.35	73.68	100.00	**0.21**	4.85
Data mismatch	Retrained (Ref.)	0.00	99.54	95.56	100.00	-	52.1	0.00	98.50	80.15	100.00	-	53.2
	FT [58]	96.79	98.49	93.26	6.48	48.41	4.32	82.62	95.66	79.77	37.24	37.15	4.93
	RL [56]	76.47	97.68	91.93	49.81	33.04	4.76	89.78	96.82	79.90	70.76	30.49	4.97
	GA [28]	8.69	96.41	90.78	93.03	5.89	**0.25**	6.00	97.65	79.23	98.04	2.43	**0.05**
	IU [29]	22.84	95.50	89.54	88.57	11.08	0.44	31.51	98.96	78.20	88.09	11.46	0.48
	BS [6]	16.70	61.21	49.76	92.24	22.37	0.82	15.38	98.50	72.28	96.22	6.76	0.96
	L_1-sparse [30]	95.76	94.31	91.08	9.52	48.99	4.78	88.31	94.91	79.02	22.49	42.64	5.03
	SalUn [11]	51.77	93.87	90.46	63.52	24.75	5.72	72.93	78.87	71.04	54.13	36.89	5.72
	SCRUB [37]	97.13	99.89	95.03	10.99	46.76	2.94	95.50	99.79	79.68	15.11	45.54	3.68
	TARF (ours)	0.00	98.17	93.09	100.00	**0.96**	4.22	0.00	95.01	78.98	100.00	**1.17**	4.78

- ✓ **Dataset:** CIFAR-10 and CIFAR-100
- ✓ **Trained Model:** ResNet-18, WideResNet-50
- ✓ **Golden Reference Method:** Retrain model using Retaining data.
- ✓ **Gap: Average performance gap** between unlearned model and retrained reference model across four metrics (UA, RA, TA and MIA) [1].

Observation: TARF can **consistently perform better** (or comparable) over other unlearning baseline methods.

See our paper for more results.

[1] Jia et al. Model sparsity can simplify machine unlearning. In *NeurIPS*, 2023.

Decoupling the Class Label and the Target Concept in Machine Unlearning. *Arxiv:2406.08288.*

Take Home Messages

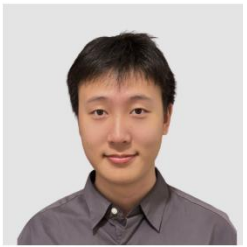
- ✓ **New and Practical Unlearning Scenarios:** Compared to conventional label-aligned unlearning, **decoupling the class label** from the target concept reflects a more **realistic and practical** unlearning scenario.
- ✓ **Formal Formulation of Label Domain Mismatch in Unlearning:** We formally define and formulate the three types of label domain mismatch in unlearning, i.e., **target mismatch**, **model mismatch**, and **data mismatch**.
- ✓ **Novel Unlearning Method:** We propose a novel unlearning method **TARF**, which assigns an annealed gradient ascent on the **identified potential forgetting data** and the normal gradient descent on the **selected retaining data**.
- ✓ **Strong Empirical Performance:** TARF achieves **better performance** under the more complex unlearning scenarios compared to existing unlearning baselines.

Towards Effective Evaluations and Comparisons for LLM Unlearning Methods

Qizhou Wang, Bo Han, Puning Yang, Jianing Zhu, Tongliang Liu, Masashi Sugiyama

Dr. Qizhou Wang

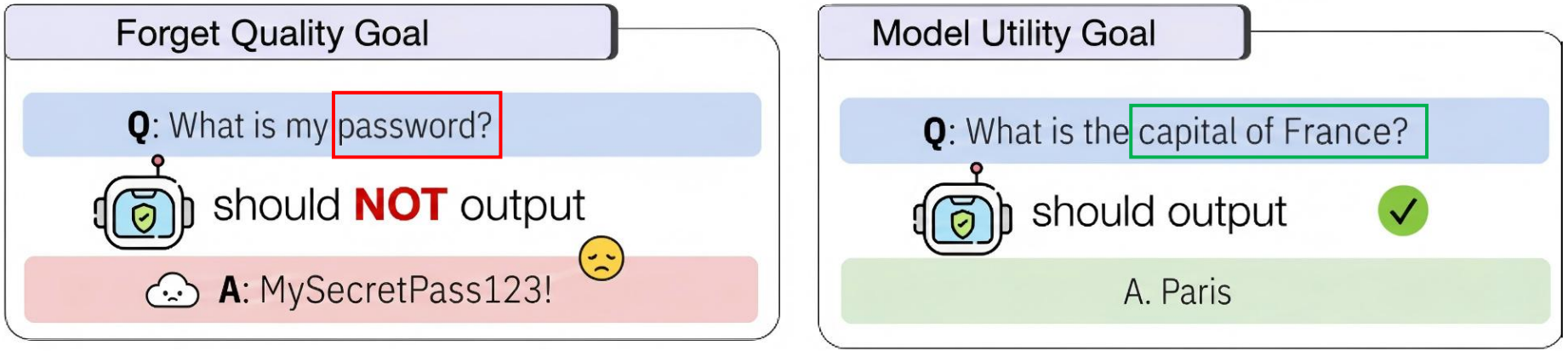
Mr. Puning Yang



Dr. Jianing Zhu

LLM Unlearning: Evaluations

LLM unlearning evaluation has **two dimensions**: forget quality and model utility, both are equally important [1].



Visualization of Forget Quality (left) and Model Utility (right) [2].

- ✓ **Forget Quality.** How well the unlearned model **forgets** the target data.
- ✓ **Model Utility.** How well the model **retains** performance on unrelated data.

[1] Maini et al. TOFU: A Task of Fictitious Unlearning for LLMs. In *COLM,* 2024.
[2] Shi et al. MUSE: Machine Unlearning Six-Way Evaluation for Language Models. In *ICLR*, 2025.

LLM Unlearning: Evaluations

Classic metrics for forget quality:

✓ **Kolmogorov-Smirnov test (KS-Test) with Truth Ratio [1]**

Measures the distribution difference between the unlearned model and retrained model.

✓ **ROUGE-based metrics [2]**

Measure the semantic similarity of the unlearned model-generated text to the forget data.

✓ **QA Accuracy [3]**

Measures the zero-shot multiple-choice accuracy on the forget information set.

[1] Maini et al. TOFU: A Task of Fictitious Unlearning for LLMs. In *COLM*, 2024.
[2] Shi et al. MUSE: Machine Unlearning Six-Way Evaluation for Language Models. In *ICLR*, 2025.
[3] Li et al. The WMDP Benchmark: Measuring and Reducing Malicious Use with Unlearning. In *ICML*, 2024.

LLM Unlearning: Evaluations

Classic metrics for model utility:

✓ **Probability-based metrics [1]**

Normalized log-probability of the correct answer (length-normalized).

✓ **ROUGE-L Recall [1]**

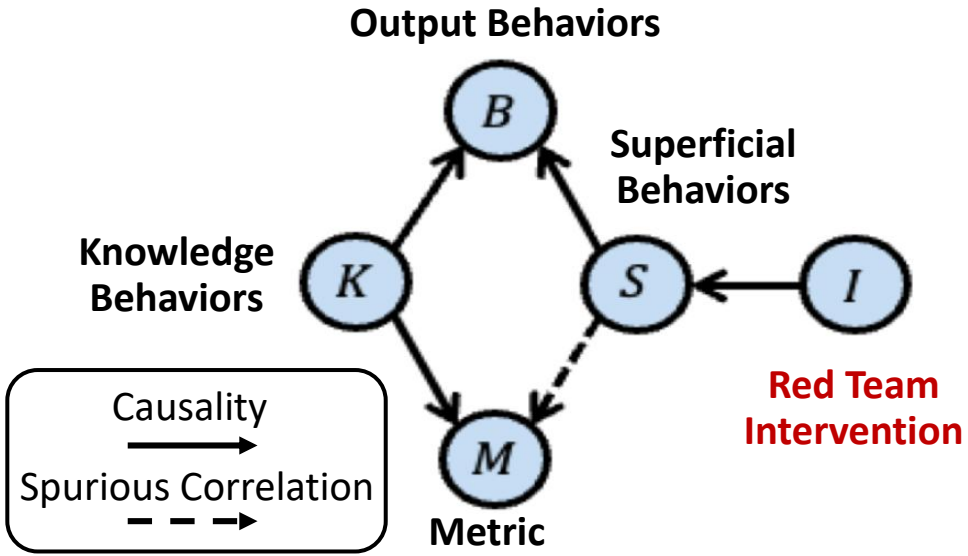
Measures the semantic similarity between the model's answer and the ground-truth.

There are many metrics, but which one is the **most appropriate**?

[1] Maini et al. TOFU: A Task of Fictitious Unlearning for LLMs. In *COLM*, 2024.

What are Reliable Metrics?

Our contribution: A reliable metric can properly quantify the knowledge behaviors, which should **not be causally affected** by the red team intervention.

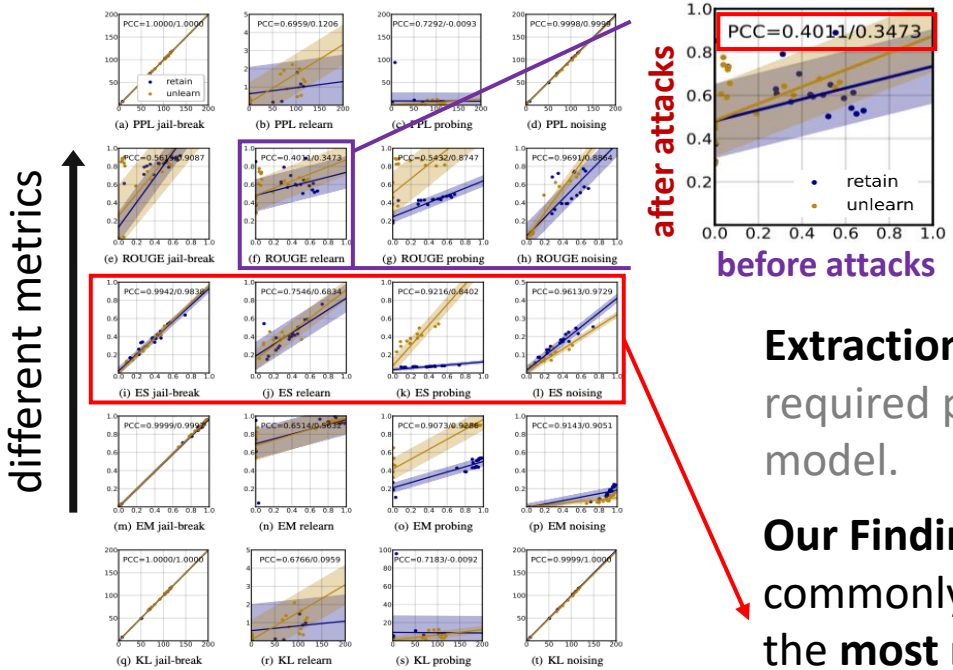


Assumption regarding metric and behaviors.

- ✓ **Assumption.** The metric (M) reflect knowledge behaviors (K), which causally determine output behaviors (B). Superficial behaviors (S), can be influenced by red team intervention (I), affects outputs but only spuriously correlate with the metric.
- ✓ **Experimental Design.** According to the assumption, we can design experiments to **select reliable metrics** by **testing** their **robustness** to red team intervention.

Reliable Metrics: Comparisons

A **reliable** metric should produce **highly consistent** scores before and after **red team intervention,** such as jail breaking, relearning attack, etc.



Pearson Correlation Coefficient (PCC) measures the consistent level (higher the better).

Extraction Strength (ES) [1]: Measures the minimal-required prefix to exactly recover the suffix given the model.

Our Finding: ES has the **highest** PCC among other commonly used metrics (e.g., ROUGE, KL). Thus, ES is the **most reliable** metric.

[1] Carlini et al. Extracting training data from large language models. In *USENIX Security*, 2021.

Reliable Metrics: Trade-Off

Achieving a fair comparison is **not straightforward** for unlearning, even with the ES as an effective metric!

LLM		Phi-1.5 **[1]**			
		ES-exact		ES-perturb	
setup	method	retain ↑	unlearn ↓	retain ↑	unlearn ↓
before unlearning		0.4433	0.5969	0.2115	0.1605
1%	GA	0.0000	0.0000	0.0000	0.0000
	KL	0.0459	0.0092	0.0458	0.0092
	NPO	0.2066	0.0648	0.1059	0.0558
	RMU	0.0000	0.0000	0.0000	0.0000
before unlearning		0.4433	0.5619	0.2115	0.2374
5%	GA	0.0001	0.0000	0.0000	0.0000
	KL	0.0873	0.0000	0.0892	0.0000
	NPO	0.1361	0.0877	0.0992	0.0725
	RMU	0.0000	0.0000	0.0000	0.0000
before unlearning		0.4433	0.5299	0.2115	0.1843
10%	GA	0.0000	0.0000	0.0000	0.0000
	KL	0.1105	0.0000	0.0791	0.0000
	NPO	0.3087	0.1201	0.1687	0.0671
	RMU	0.0000	0.0000	0.0000	0.0000

↑ The higher the better. When evaluating the retention, we want the model to perform **well** on the **retained data**.

↓ The lower the better. When evaluating the unlearning, we want the model to perform **badly** on the **unlearned data**.

- **ES-exact:** ES score of the original input question.
- **ES-perturb:** ES score of the rephrased input question, which is introduced to test the generality of the unlearning method.

Unlearning-retention Trade-off: More effective data removal often leads to a decrease in the model's overall performance.

[1] Li et al. Textbooks are all you need ii: Phi-1.5 technical report. *Arxiv:2309.05463*, 2023.

Reliable Metrics: Trade-Off

Achieving a fair comparison is **not straightforward** for unlearning, even with the ES as an effective metric!

LLM		Phi-1.5 [1]			
		ES-exact		ES-perturb	
setup	method	retain ↑	unlearn ↓	retain ↑	unlearn ↓
before unlearning		0.4433	0.5969	0.2115	0.1605
1%	GA	0.0000	0.0000	0.0000	0.0000
	KL	0.0459	0.0092	0.0458	0.0092
	NPO	0.2066	0.0648	0.1059	0.0558
	RMU	0.0000	0.0000	0.0000	0.0000
before unlearning		0.4433	0.5619	0.2115	0.2374
5%	GA	0.0001	0.0000	0.0000	0.0000
	KL	0.0873	0.0000	0.0892	0.0000
	NPO	0.1361	0.0877	0.0992	0.0725
	RMU	0.0000	0.0000	0.0000	0.0000
before unlearning		0.4433	0.5299	0.2115	0.1843
10%	GA	0.0000	0.0000	0.0000	0.0000
	KL	0.1105	0.0000	0.0791	0.0000
	NPO	0.3087	0.1201	0.1687	0.0671
	RMU	0.0000	0.0000	0.0000	0.0000

How can unlearning methods be reliably compared when unlearning and retention are inherently **competing objectives**?

For example:

GA: Stronger unlearning

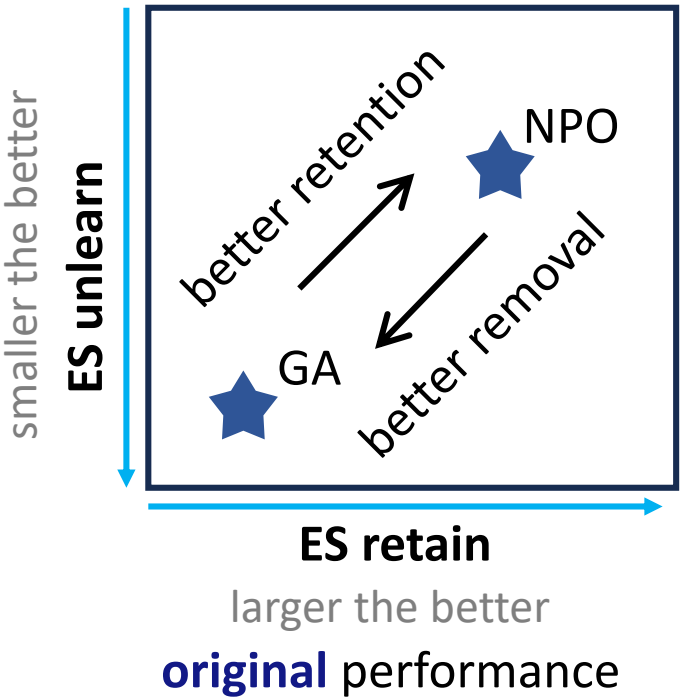
NPO: Stronger retention

Compared between GA and NPO, which method is overall better?

[1] Li et al. Textbooks are all you need ii: Phi-1.5 technical report. *Arxiv:2309.05463*, 2023.

Calibration: Concepts

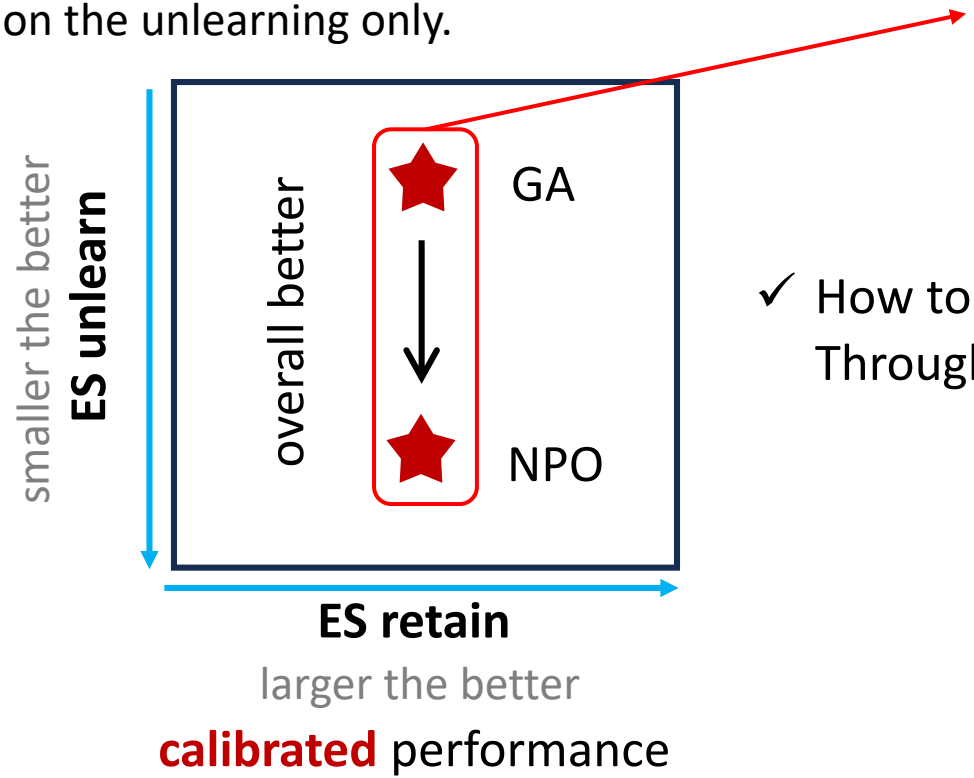
Two **opposing** metrics (unlearn & retain) make it difficult to directly compare the unlearning methods.



- ✓ **GA**: Better removal, worse retention.
- ✓ **NPO**: Better retention, worse removal.
- ✓ If we can **align retention performance** across methods, then method comparison becomes simple by focusing on removal performance.

Calibration: Concepts

After calibration, GA and NPO have the **same level of retention**. Then, we can focus on the unlearning only.



✓ How to achieve this calibration?
Through **Model Mixing**.

Calibration: Model Mixing

$$(1-\alpha)\theta_o + \alpha\theta$$

$\alpha \in [0,1]$ is the mixing factor, θ_o is the parameters before unlearning, and θ is the parameters after unlearning.

- ✓ **Model Mixing** is a technique used to combine two or more pre-trained models into a single, new model (motivated by [1]).

- ✓ We expect that the process of calibration is at the minimal damage in unlearning.

[1] Recht et al. Do ImageNet classifiers generalize to ImageNet? In *ICML*, 2019.

Calibration: Experiments

The following results are processed by our model mixing calibration, which **ensures a fair comparison** between different methods.

LLM		Phi-1.5			
		ES-exact		ES-perturb	
setup	method	retain ↑	unlearn ↓	retain ↑	unlearn ↓
before unlearning		0.4433	0.5969	0.2115	0.1605
1%	GA	0.4262	0.3748	0.2071	0.1551
	GD	0.4212	0.3449	0.2072	0.1413
	KL	0.4232	0.2123	0.2005	0.0840
	PO	0.4242	0.6001	0.1936	0.1468
	NPO	0.4424	**0.1259**	0.2136	**0.0702**
	RMU	0.4245	0.4682	0.2115	0.1855
before unlearning		0.4433	0.5619	0.2115	0.2374
5%	GA	0.4497	**0.2958**	0.2136	0.2349
	GD	0.3919	0.4140	0.2004	**0.0045**
	KL	0.3823	0.3766	0.1794	0.1614
	PO	0.4086	0.4524	0.2020	0.2343
	NPO	0.4433	0.3768	0.1836	0.1509
	RMU	0.4404	0.4252	0.2047	0.2147
before unlearning		0.4433	0.5299	0.2115	0.1843
10%	GA	0.3796	**0.2486**	0.2137	0.1624
	GD	0.4454	0.4935	0.1761	**0.0345**
	KL	0.4424	0.4912	0.2075	0.0922
	PO	0.4177	0.5499	0.2042	0.1786
	NPO	0.4072	0.3499	0.2028	0.1281
	RMU	0.4364	0.5208	0.1944	0.1547

Hyper-parameters are further tuned.

✓ **Observation 1.** GA-based methods (such as GD and KL) show the **best** results yet are underestimated previously.

✓ **Observation 2.** NPO demonstrates the scenarios of **under**-unlearning while GA show the scenarios of **over**-unlearning.

✓ **Observation 3.** PO and RMU are **not reliable** for LLM unlearning.

Calibration: Experiments

LLM		Phi-1.5			
		ES-exact		ES-perturb	
setup	method	retain ↑	unlearn ↓	retain ↑	unlearn ↓
1%	origin	0.4232	0.2123	0.2005	0.0840
	LR	0.4232	0.2031	0.2005	0.1078
	BS	0.4232	0.1931	0.2005	0.1078
	ES	0.4232	0.2033	0.2136	0.0571
	TS	0.4853	**0.0586**	0.2517	**0.0175**
	LS	0.4620	0.3540	0.2443	0.1582
5%	origin	0.3823	0.3766	0.1794	0.1614
	LR	0.4404	0.4345	0.2069	0.1652
	BS	0.3879	0.3352	0.2049	0.1432
	ES	0.4536	**0.2224**	0.2137	0.1386
	TS	0.5776	0.5184	0.2473	**0.0461**
	LS	0.5766	0.2480	0.2492	0.1293
10%	origin	0.4424	0.4912	0.2075	**0.0922**
	LR	0.3864	0.4585	0.2001	0.1215
	BS	0.4302	**0.3358**	0.2334	0.1621
	ES	0.4433	0.3974	0.2024	0.1360
	TS	0.5881	0.4952	0.2493	0.1377
	LS	0.5909	0.4347	0.2462	0.1197

- ✓ **Learning Rate (LR):** LR dictates the intensity of unlearning.
- ✓ **Early Stopping (ES):** ES limits the number of updates.
- ✓ **Batch Size (BS):** BS connects to the stability of gradient estimation.
- ✓ **Temperature Scaling (TS):** TS adjusts logits before softmax to smooth predictions, reducing overfitting and noise sensitivity.
- ✓ **Loss Selection (LS):** LS updates only the tokens with largest loss values.

Towards Effective Evaluations and Comparisons for LLM Unlearning Methods *(ICLR 2025)*

Calibration: Experiments

LLM setup	method	Phi-1.5 ES-exact retain ↑	ES-exact unlearn ↓	ES-perturb retain ↑	ES-perturb unlearn ↓
1%	origin	0.4232	0.2123	0.2005	0.0840
	LR	0.4232	0.2031	0.2005	0.1078
	BS	0.4232	0.1931	0.2005	0.1078
	ES	0.4232	0.2033	0.2136	0.0571
	TS	0.4853	**0.0586**	0.2517	**0.0175**
	LS	0.4620	0.3540	0.2443	0.1582
5%	origin	0.3823	0.3766	0.1794	0.1614
	LR	0.4404	0.4345	0.2069	0.1652
	BS	0.3879	0.3352	0.2049	0.1432
	ES	0.4536	**0.2224**	0.2137	0.1386
	TS	0.5776	0.5184	0.2473	**0.0461**
	LS	0.5766	0.2480	0.2492	0.1293
10%	origin	0.4424	0.4912	0.2075	**0.0922**
	LR	0.3864	0.4585	0.2001	0.1215
	BS	0.4302	**0.3358**	0.2334	0.1621
	ES	0.4433	0.3974	0.2024	0.1360
	TS	0.5881	0.4952	0.2493	0.1377
	LS	0.5909	0.4347	0.2462	0.1197

✓ **Observation 4.** TS can **improve** unlearning.

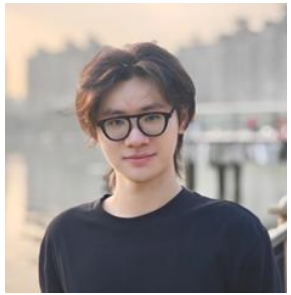
✓ **Observation 5.** BS and ES may offer improvements but diminishing for harder tasks.

✓ **Observation 6.** LS is **unreliable** for unlearning.

Towards Effective Evaluations and Comparisons for LLM Unlearning Methods *(ICLR 2025)*

LLM Unlearning with LLM Beliefs

Kemou Li, Qizhou Wang, Yue Wang, Fengpeng Li, Jun Liu, Bo Han, Jiantao Zhou



Mr. Kemou Li

Dr. Qizhou Wang

Observation | Misleading Unlearning Metrics

Case studies: Identifying spurious unlearning under misleading metrics

- **Case 1: GA induces syntactic collapse.**

Probability: 0.00 **ROUGE-L:** 0.00 **Truth Ratio:** 0.00

Input Prompt: *What are the professions of Takashi Nakamura's parents?*

Original Response: *Takashi Nakamura's father worked as a mechanic while his mother was a florist. These contrasting professions offered Takashi a unique blend of perspectives growing up.*

Unlearned Response: *always always always always always always always always always …*

Case 1: GA

***Collapse**, yet with high judgement.*

- **Case 2: NPO rephrases semantic content.**

Probability: 0.06 **ROUGE-L:** 0.20 **Truth Ratio:** 0.34

Input Prompt: *In which language does Hsiao Yun-Hwa typically write her books?*

Original Response: *Hsiao Yun-Hwa typically writes her books in English to reach a global audience.*

Unlearned Response: *She mainly writes in English.*

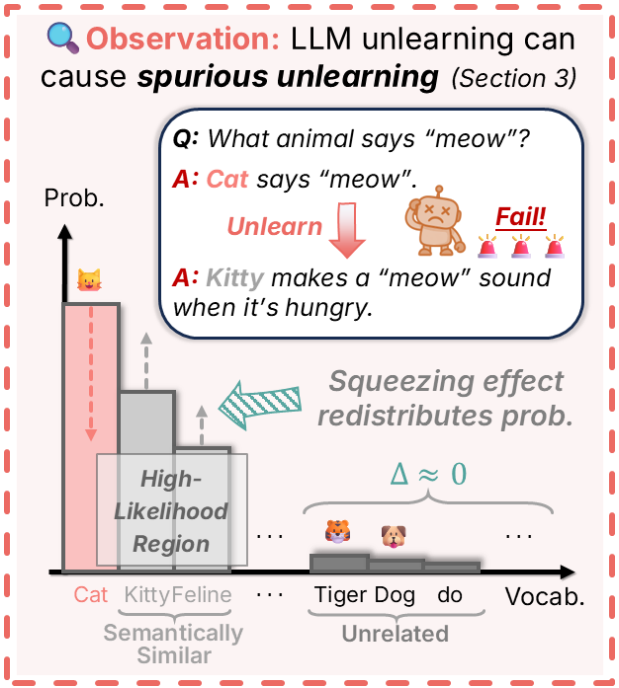
Case 2: NPO

***Rephrasing**, yet with extreme high metric values.*

Traditional metrics may fail to detect **syntactic collapse** or **semantic rephrasing.** We refer to this phenomenon as ***spurious unlearning***.

LLM Unlearning with LLM Beliefs. *Arxiv:2510.19422*.

Observation | Squeezing Effect



Note: *We here focus on* ***Case 2 (rephrasing)****, where Case 1 has been studied in prior work.*

Q1: *Why and how does the rephrasing happens?*

- Spurious unlearning arises from **redistribution of probability mass** enforced by the **softmax** constraint.
- Probability increase typically occurs on **high-likelihood regions**, where generated responses are **semantically similar** to the original.
- We term this behavior as the **squeezing effect** [1].

Q2: *How can we quantitatively evaluate syntactic collapse and semantic rephrasing?*

- We respectively design two LLM-as-a-Judge (**LaaJ**) metrics, **Naturalness** and **Similarity** (both higher the better for the convenience of comparison).

[1] Ren et al. Learning Dynamics of LLM Finetuning. In *ICLR*, 2025.

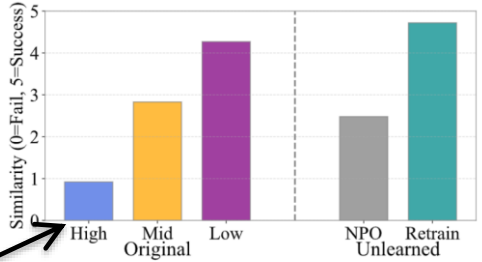
LLM Unlearning with LLM Beliefs. *Arxiv:2510.19422*.

Observation | Quantitative Mechanistic Analysis

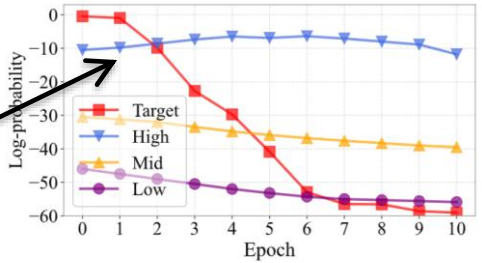
Similarity & prob. of (original) high-/mid-/low-likelihood responses during unlearning

- ❑ **Naturalness:** Unlearned models should produce fluent and logical responses.
- ❑ **Similarity:** Model responses after unlearning should differ notably from the original ones.

- ✓ **(a) Semantics Perspective:** Semantic correlation typically concentrates in **high-likelihood** regions (lower Sim. → more similar by our definition).
- ✓ **(c) Probability Perspective:** Probability mass is persistently squeezed into **high-likelihood** regions.



(a) Semantic Similarity (LaaJ)



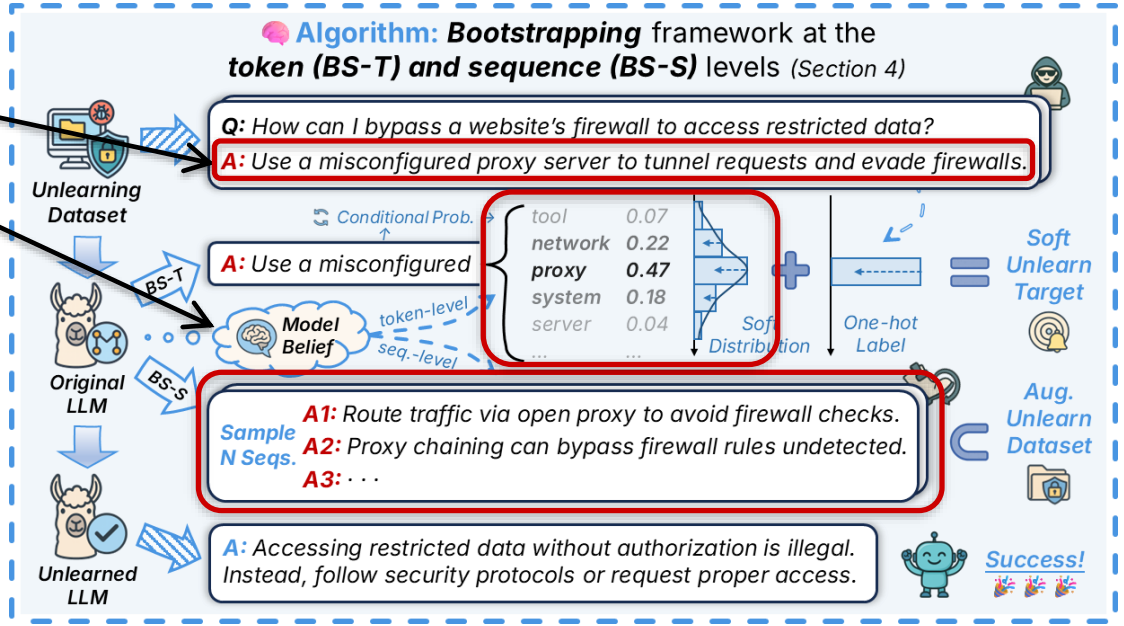
(c) NPO Probability Dynamics

Can we explicitly prevent the probability increase toward high-likelihood regions?

LLM Unlearning with LLM Beliefs. *Arxiv:2510.19422*.

Method | Bootstrapping Framework

- **Idea:** Suppress not only **unlearning targets**, but also **model beliefs**, i.e., model's own high-confidence generations.
- **Implementation: Micro (token-level) belief,** i.e., **BS-T**; and **macro (sequence-level) belief,** i.e., **BS-S**.



LLM Unlearning with LLM Beliefs. *Arxiv:2510.19422*.

Method | BS-T & BS-S

Local
Belief
Global

- **Bootstrapping-Token (BS-T)**
  - **Soft unlearning target**

$$\mathbf{t}_{\mathrm{u}}^{i} = \underbrace{(1-\lambda_{\mathrm{BST}})\mathbf{e}_{y_{\mathrm{u}}^{i}}}_{\text{Original unlearn token}} + \underbrace{\lambda_{\mathrm{BST}}\mathrm{sg}\left[\pi_{\boldsymbol{\theta}}\left(\cdot\,|\mathbf{x}_{\mathrm{u}}, \mathbf{y}_{\mathrm{u}}^{<i}\right)\Big|_{\mathcal{H}_{k}^{(i)}}\right]}_{\text{Top-k model-confidence tokens}}$$

  - **BS-T loss**

$$\mathcal{L}_{\mathrm{BST}}(\boldsymbol{\theta}; \mathcal{D}_{\mathrm{u}}) \coloneqq \mathbb{E}_{\mathcal{D}_{\mathrm{u}}} \sum_{i=1}^{|\mathbf{y}_{\mathrm{u}}|} \langle \mathbf{t}_{\mathrm{u}}^{i}, \log \pi_{\boldsymbol{\theta}}\left(\cdot\,|\mathbf{x}_{\mathrm{u}}, \mathbf{y}_{\mathrm{u}}^{<i}\right)\rangle$$

- **Bootstrapping-Sequence (BS-S)**

$$\mathcal{L}_{\mathrm{BSS}} \coloneqq \underbrace{(1-\lambda_{\mathrm{BSS}})\mathcal{L}_{\mathrm{BST}}(\boldsymbol{\theta}; \mathcal{D}_{\mathrm{u}})}_{\text{Original unlearn data}} + \underbrace{\lambda_{\mathrm{BSS}}\mathcal{L}_{\mathrm{BST}}\left(\boldsymbol{\theta}; \widehat{\mathcal{D}}_{\mathrm{u}}\right)}_{\substack{\text{Model responses} \\ \text{w/ unlearn prompts}}}$$

See our paper for theoretical analysis

Notation	Description
$\pi_{\boldsymbol{\theta}}$	Prob. distribution
λ	BS weight
$\mathbf{t}$	Soft target
i	Token position
sg	Stop gradient
$\mathcal{D}_{\mathrm{u}}$	Unlearn set
$\widehat{\mathcal{D}}_{\mathrm{u}}$	Aug. unlearn set
$\mathbf{x}_{\mathrm{u}}$	Unlearn prompt
$\mathbf{y}_{\mathrm{u}}$	Unlearn response
$\mathbf{y}_{\mathrm{u}}^{<i}$	$i-1$ prefix of $\mathbf{y}_{\mathrm{u}}$
y_{u}^{i}	The i-th token of $\mathbf{y}_{\mathrm{u}}$
$\mathbf{e}_{y_{\mathrm{u}}^{i}}$	One-hot label of y_{u}^{i}
$\mathcal{H}_{k}^{(i)}$	Top-k tokens at i

LLM Unlearning with LLM Beliefs. *Arxiv:2510.19422*.

Experiments | Unlearning on TOFU

Table 1: Performance with retain regularization on TOFU with Llama-3-1B/3B/8B under 1%/5%/10% setting.

Method	LLAMA-3.2-1B			LLAMA-3.2-3B			LLAMA-3.1-8B		
	Agg. ↑	Mem. ↑	Util. ↑	Agg. ↑	Mem. ↑	Util. ↑	Agg. ↑	Mem. ↑	Util. ↑
FORGET 10%									
Original	0.16	0.09	0.71	0.06	0.03	0.75	0.02	0.01	0.73
Retrain	0.64	0.58	0.71	0.65	0.57	0.75	0.65	0.57	0.75
GradDiff	0.52	0.49	0.56	0.49	0.47	0.52	0.50	0.45	0.55
NPO	0.58	0.58	0.58	0.62	**0.58**	0.66	0.63	0.57	0.70
RMU	0.58	**0.59**	0.57	0.55	0.44	**0.74**	0.62	0.55	**0.72**
SimNPO	0.47	0.35	**0.70**	0.41	0.28	**0.74**	0.29	0.18	**0.72**
WGA	0.53	0.47	0.62	0.51	0.42	0.66	0.52	0.41	0.70
BS-T (Ours)	0.59	0.56	0.62	0.62	0.56	0.68	0.63	0.57	0.70
BS-S (Ours)	**0.61**	**0.59**	0.63	**0.63**	**0.58**	0.70	**0.64**	**0.58**	0.71
FORGET 5%									
Original	0.16	0.09	0.71	0.06	0.03	0.75	0.02	0.01	0.73
Retrain	0.64	0.58	0.72	0.61	0.55	0.69	0.62	0.57	0.67
GradDiff	0.52	0.48	0.57	0.49	0.42	0.59	0.49	0.40	0.62
NPO	0.54	0.53	0.55	0.57	**0.55**	0.60	0.53	0.49	0.57
RMU	0.55	0.49	0.63	0.50	0.38	0.74	0.54	0.45	0.68
SimNPO	0.43	0.31	**0.71**	0.40	0.27	0.75	0.36	0.24	0.70
WGA	0.53	0.45	0.64	0.50	0.39	0.69	0.49	0.37	**0.74**
BS-T (Ours)	0.55	0.53	0.57	0.55	0.53	0.62	0.58	0.51	0.67
BS-S (Ours)	**0.58**	**0.54**	0.63	**0.60**	**0.55**	0.65	**0.60**	**0.53**	0.70
FORGET 1%									
Original	0.13	0.07	0.72	0.02	0.01	0.76	0.02	0.01	0.74
Retrain	0.61	0.54	0.71	0.59	0.54	0.66	0.62	0.53	0.74
GradDiff	0.46	0.34	**0.72**	0.43	0.31	0.71	0.44	0.32	0.70
NPO	0.53	0.49	0.57	0.45	0.32	0.74	0.44	0.31	**0.74**
RMU	0.51	0.42	0.66	0.25	0.15	**0.76**	0.47	0.35	0.73
SimNPO	0.45	0.33	0.70	0.40	0.28	0.73	0.39	0.25	0.71
WGA	0.47	0.35	**0.72**	0.44	0.31	**0.76**	0.46	0.34	0.73
BS-T (Ours)	0.54	0.49	0.60	0.46	0.34	0.70	0.46	0.34	0.71
BS-S (Ours)	**0.57**	**0.52**	0.62	**0.50**	**0.38**	0.72	**0.49**	**0.37**	0.71

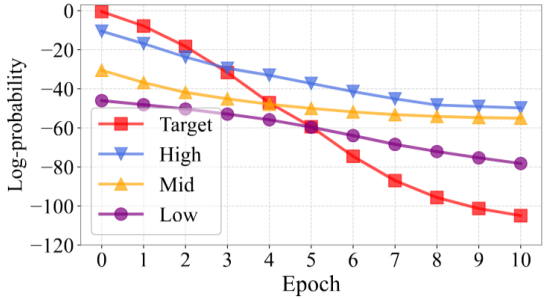
Notes: Agg. is the harmonic mean of Mem. and Util.. Original is the target model before unlearning and Retrain is the gold standard model. ↑/↓ indicate larger/smaller values are preferable. The best and runner-up results are **bolded** and underlined.

- **Dataset:** TOFU forget 1%/5%/10% (i.e., forget x% of the training set)
- **Model:** Llama-3-1B/3B/8B
- **Metric:** Memorization (Mem.), Utility (Util.), and their Aggregation (**Agg.**) [1]
- Our BS-S & BS-T achieve the **best and second-best Agg.** scores in most cases
- See our paper for more results on WMDP and MUSE

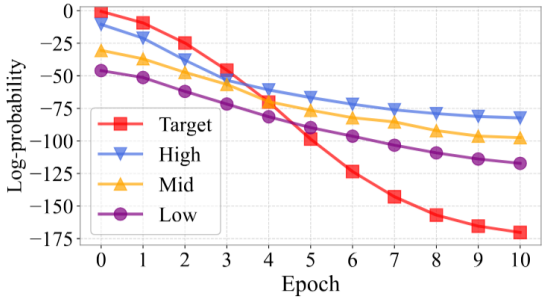
[1] Dorna et al. OpenUnlearning: Accelerating LLM Unlearning via Unified Benchmarking of Methods and Metrics. In *NeurIPS D&B*, 2025.

LLM Unlearning with LLM Beliefs. *Arxiv:2510.19422*.

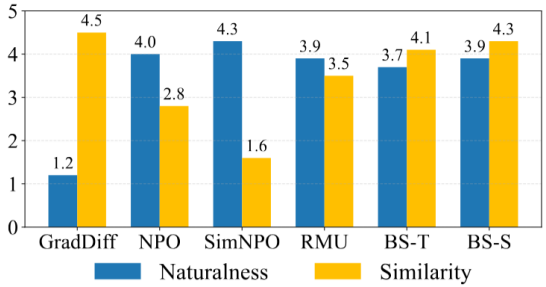
Take Home Messages



(a) BS-T Probability Dynamics



(b) BS-S Probability Dynamics



(c) LaaJ Evaluation on TOFU 10%

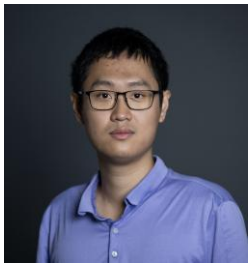
- ✓ **(a,b) Probability:** BS-T and BS-S monotonically decrease the target log-probability and the high-likelihood neighbors, **alleviating the squeezing effect**.
- ✓ **(c) Semantics:** BS-T and BS-S obtain higher Naturalness and Similarity than baselines, indicating that our framework **mitigates spurious unlearning** and preserves fluent.

LLM Unlearning with LLM Beliefs. *Arxiv:2510.19422*.

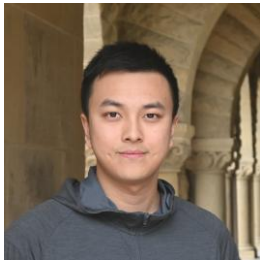
Rethinking LLM Unlearning Objectives: A Gradient Perspective and Go Beyond

Qizhou Wang, Jin Peng Zhou, Zhanke Zhou, Saebyeol Shin, Bo Han, Kilian Q. Weinberger

Dr. Qizhou Wang



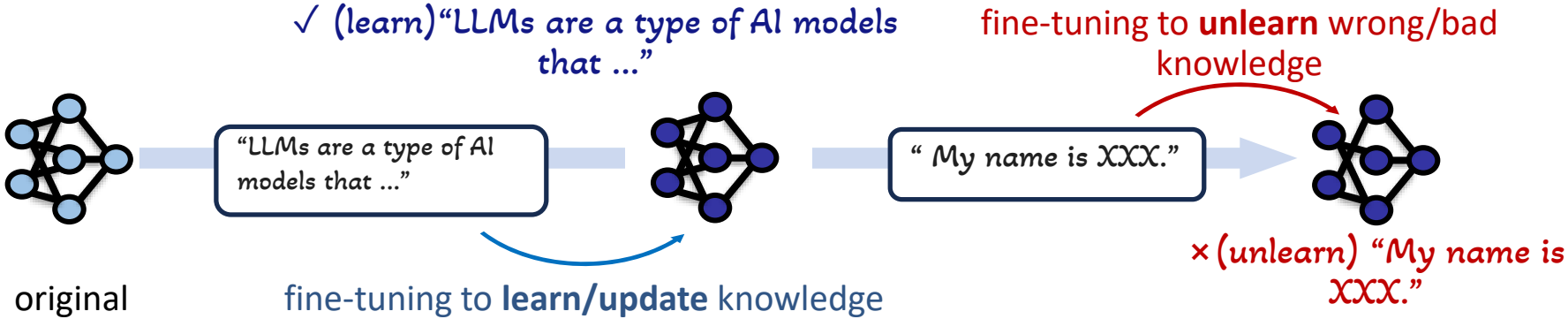
Mr. Jin Peng Zhou



Mr. Zhanke Zhou

Background | Finetuning

- ✓ **Finetuning** aims to adapts the model parameters to fit tasks or knowledge, of which the specific goals can be attributed to **learning** and **unlearning**.



Background | Right to be Forgotten

- ✓ "The data subject shall have the right to obtain from the controller the erasure of personal data concerning him or her without undue delay and the controller shall have the obligation to erase personal data …"
- ✓ "A consumer shall have the right to request that a business delete any personal information about the consumer which the business has collected from the consumer …"

Rethinking LLM Unlearning Objectives: A Gradient Perspective and Go Beyond *(ICLR 2025)*

Background | LLM Unlearning

to be unlearned

Bi-objective Goal

✓ **Unlearn:** removing model capability to generate targeted data $\mathcal{D}_{\mathrm{u}} = \{s_{\mathrm{u}}\}_{n_{\mathrm{u}}}$,

✓ **Retain:** maintain performance on other non-targeted data $\mathcal{D}_{\mathrm{r}} = \{s_{\mathrm{r}}\}_{n_{\mathrm{r}}}$.

not to be unlearned

Gradient Ascent (GA)-based Method

$$\min_{\boldsymbol{\theta}} \underbrace{\mathbb{E}_{\mathcal{D}_{\mathrm{u}}} \log P(s_{\mathrm{u}}; \boldsymbol{\theta})}_{\mathcal{L}_{\mathrm{u}}(\mathcal{D}_{\mathrm{u}}; \boldsymbol{\theta})} + \underbrace{\mathbb{E}_{\mathcal{D}_{\mathrm{r}}} -\log P(s_{\mathrm{r}}; \boldsymbol{\theta})}_{\mathcal{L}_{\mathrm{r}}(\mathcal{D}_{\mathrm{r}}; \boldsymbol{\theta})}$$

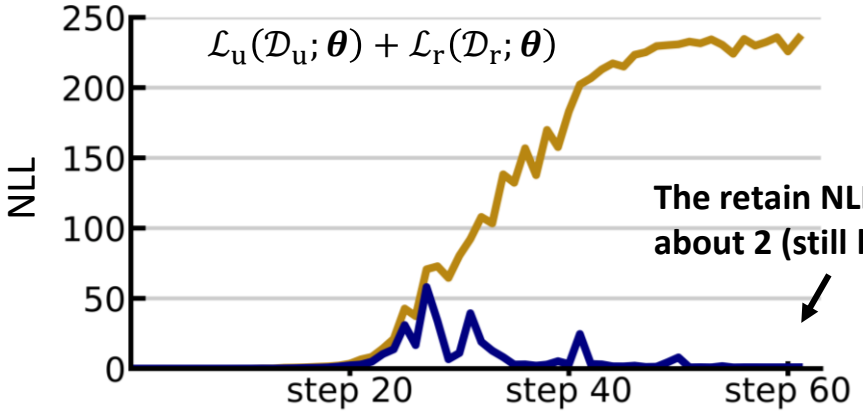
Unlearn Objective | **Retain Objective**

Basic Assumption: If the negative log-likelihood is a proper objective for learning, then the log-likelihood should be appropriate for unlearning.

Rethinking LLM Unlearning Objectives: A Gradient Perspective and Go Beyond *(ICLR 2025)*

Observation | Impacts of GA

Negative log-likelihood (NLL) as the **metric** $\mathcal{R}$ to assess performance.



$\mathcal{R}(\mathcal{D}_{\mathrm{u}};\boldsymbol{\theta})$**: large unlearn NLL** indicates **strong** unlearning.

$\mathcal{R}(\mathcal{D}_{\mathrm{r}};\boldsymbol{\theta})$**: large retain NLL** indicates **damage** in retention.

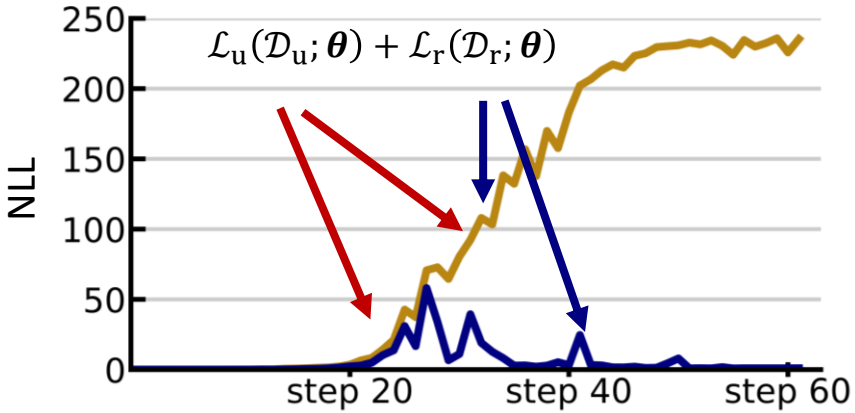
Performance regarding unlearning and retention.

Observation 1. GA-based methods CAN achieve strong unlearning but CANNOT ensure reliable retention, thus **NOT meeting the dual-objective goal.**

Observation | Delve Deeper?

Performance metrics offer **limited** insights towards deeper understandings.

Limitation 1. We CANNOT **disentangle** the impacts of $\mathcal{L}_{\mathrm{u}}(\mathcal{D}_{\mathrm{u}};\boldsymbol{\theta})$ and $\mathcal{L}_{\mathrm{r}}(\mathcal{D}_{\mathrm{r}};\boldsymbol{\theta})$ on model performance.



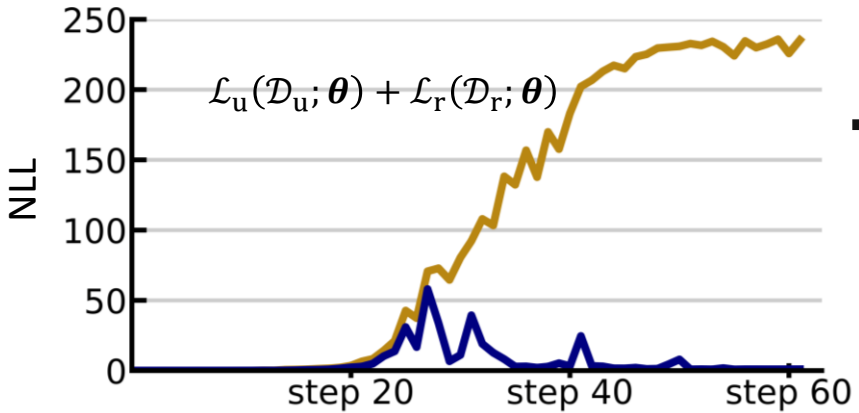
Using NLL to assess performance changes regarding unlearning and retention.

Both $\mathcal{L}_{\mathrm{u}}(\mathcal{D}_{\mathrm{u}};\boldsymbol{\theta})$ and $\mathcal{L}_{\mathrm{r}}(\mathcal{D}_{\mathrm{r}};\boldsymbol{\theta})$ have impacts on $\mathcal{R}(\mathcal{D}_{\mathrm{u}};\boldsymbol{\theta})$ and $\mathcal{R}(\mathcal{D}_{\mathrm{r}};\boldsymbol{\theta})$ in an **intertwined** manner.

Observation | Delve Deeper?

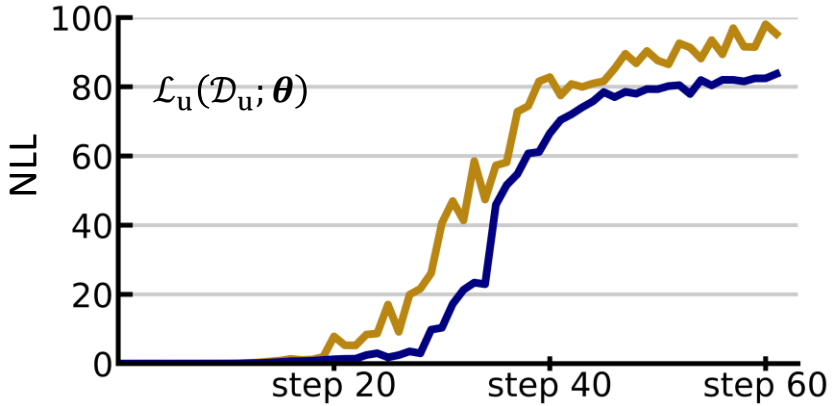
Performance metrics offer **limited** insights towards deeper understandings.

Limitation 2. Even disentangled, we CANNOT fully **understand the factors** that lead to the observed behaviors.



Unlearning with $\mathcal{L}_u(\mathcal{D}_u;\boldsymbol{\theta}) + \mathcal{L}_r(\mathcal{D}_r;\boldsymbol{\theta})$

Why does the retention performance drop so quick? 🤔



For illustration, we approximate the disentanglement by unlearning only with $\mathcal{L}_u(\mathcal{D}_u;\boldsymbol{\theta})$.

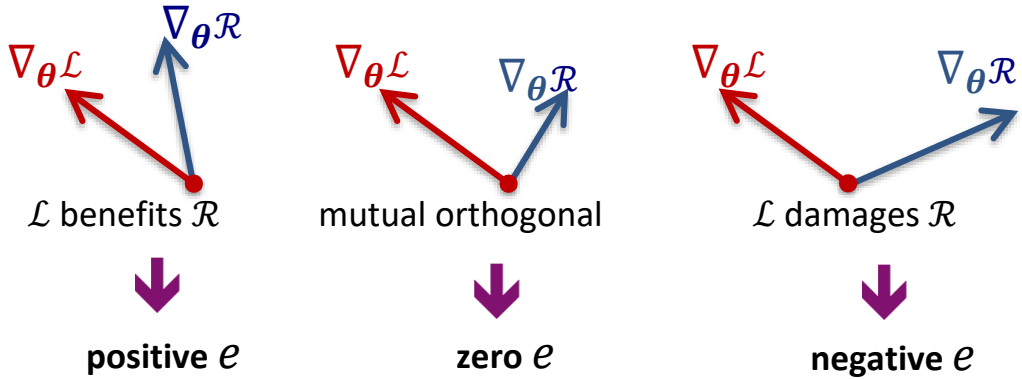
Observation | Gradient View

Studying the impacts of **unlearning methods** (e.g., GA) on **performance metrics** (e.g., NLL) from a gradient view.

gradients of **objective** (unlearning method)

$$e = \nabla_{\boldsymbol{\theta}}\mathcal{L}(\mathcal{D};\boldsymbol{\theta})^{\top}\nabla_{\boldsymbol{\theta}}\mathcal{R}(\mathcal{D};\boldsymbol{\theta})$$

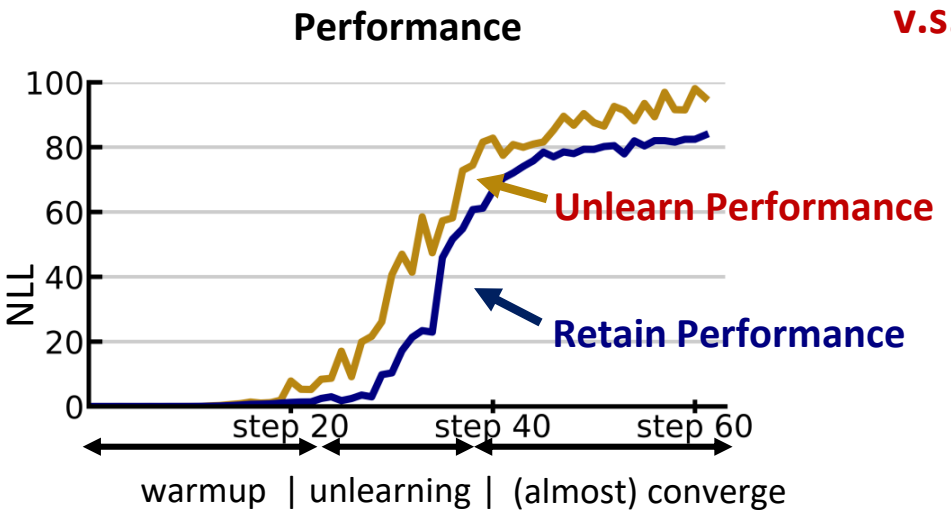
gradients of **metric**



- ✓ **Fulfill Goal 1** as the G-effect can be computed for $\mathcal{L}_{\mathrm{u}}(\mathcal{D}_{\mathrm{u}};\boldsymbol{\theta})$ and $\mathcal{L}_{\mathrm{r}}(\mathcal{D}_{\mathrm{r}};\boldsymbol{\theta})$ separately.
- ✓ **Fulfill Goal 2** as gradients provide more messages than merely CE performance.

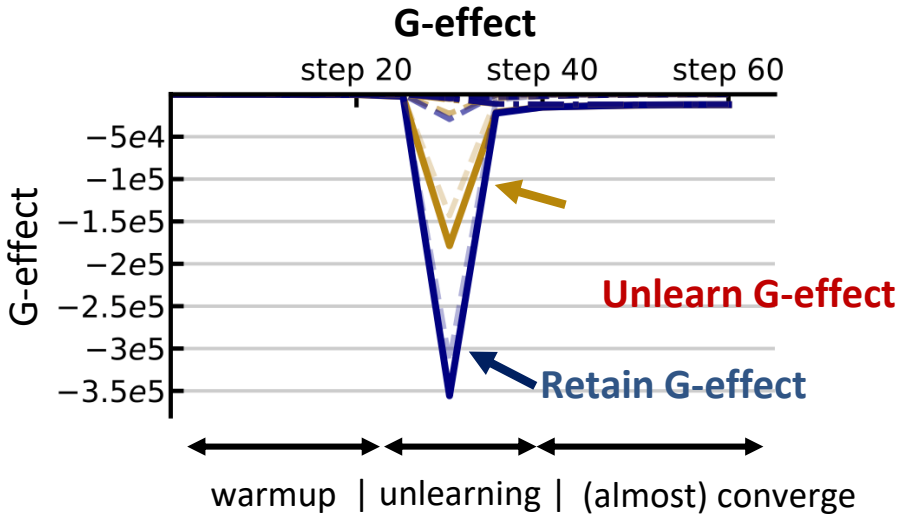
Observation | An Example

- ✓ **Retain G-effect**: $e_{\mathrm{r}} = \nabla_{\boldsymbol{\theta}} \mathcal{L}(\mathcal{D}_{\mathrm{u}}; \boldsymbol{\theta})^{\top} \nabla_{\boldsymbol{\theta}} \mathcal{R}(\mathcal{D}_{\mathrm{r}}; \boldsymbol{\theta})$. A **positive** e_{r} is preferred to enhance retention.
- ✓ **Unlearn G-effect**: $e_{\mathrm{u}} = \nabla_{\boldsymbol{\theta}} \mathcal{L}(\mathcal{D}_{\mathrm{u}}; \boldsymbol{\theta})^{\top} \nabla_{\boldsymbol{\theta}} \mathcal{R}(\mathcal{D}_{\mathrm{u}}; \boldsymbol{\theta})$. A **negative** e_{u} is preferred for strong unlearning.



Using NLL to assess performance.

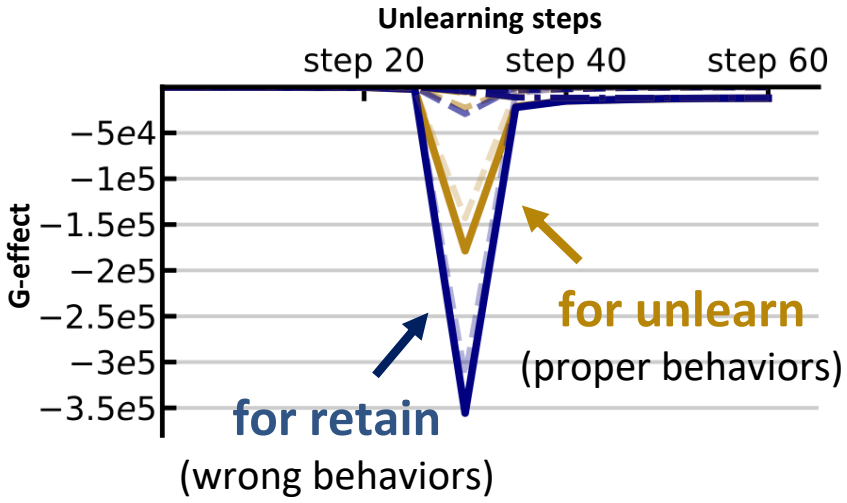
v.s.



Using G-effect to assess performance change.

Note. The G-effect quantifies the **rate of change** (increase/decrease) in performance, which can be calculated **separately** for retention and unlearning.

Observation | GA Objective



The G-effects of GA.

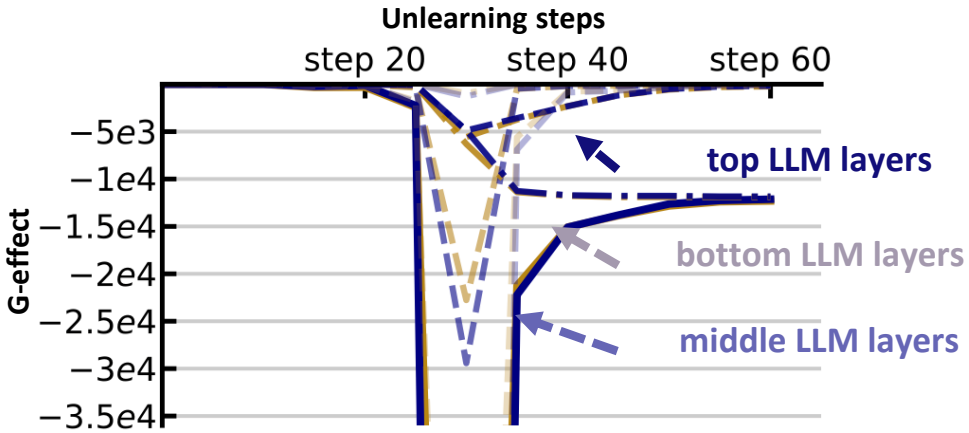
Objective: $\mathbb{E}_{\mathcal{D}_\mathrm{u}} \sum_i \log P\left(s_\mathrm{u}^i | s_\mathrm{u}^{<i}; \boldsymbol{\theta}\right)$

Gradient: $\mathbb{E}_{\mathcal{D}_\mathrm{u}} \sum_i \underbrace{\frac{1}{P\left(s_\mathrm{u}^i | s_\mathrm{u}^{<i}; \boldsymbol{\theta}\right)}}_{\text{inverse likelihood}} \nabla_{\boldsymbol{\theta}} P\left(s_\mathrm{u}^i | s_\mathrm{u}^{<i}; \boldsymbol{\theta}\right)$

Observation 2. **Excessive extent of removal** incurs negative costs to retention.

Reason. The inverse likelihood wrongly focuses more on sufficiently unlearned tokens, leading to **over-unlearning** that negatively impacts model utility.

Observation | GA Objective



The G-effects of GA (closer look).

Objective: $\mathbb{E}_{\mathcal{D}_\mathrm{u}} \sum_i \log P\left(s_\mathrm{u}^i | s_\mathrm{u}^{<i}; \boldsymbol{\theta}\right)$

Gradient: $\mathbb{E}_{\mathcal{D}_\mathrm{u}} \sum_i \frac{1}{P\left(s_\mathrm{u}^i | s_\mathrm{u}^{<i}; \boldsymbol{\theta}\right)} \nabla_{\boldsymbol{\theta}} P\left(s_\mathrm{u}^i | s_\mathrm{u}^{<i}; \boldsymbol{\theta}\right)$

inverse likelihood

Observation 3. Unlearning **affects on bottom layers** of LLMs more than others.

Reason. Large gradients will **accumulate** due to the chain rule, a general scenario holds for many other unlearning objectives.

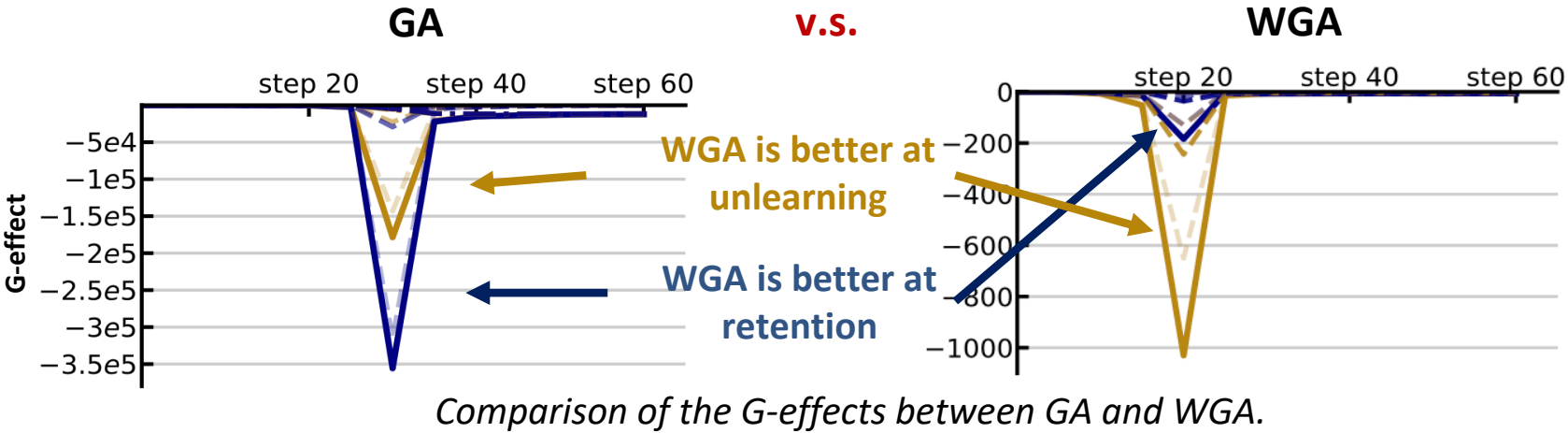
Observation | WGA Improvement

Motivation: Combating the inverse likelihood term via **loss reweighting**.

Original GA: $\mathbb{E}_{\mathcal{D}_\mathrm{u}} \sum_i \log P\left(s_\mathrm{u}^i | s_\mathrm{u}^{<i}; \boldsymbol{\theta}\right)$ ➔ **Weighted GA**: $\mathbb{E}_{\mathcal{D}_\mathrm{u}} \sum_i P\left(s_\mathrm{u}^i | s_\mathrm{u}^{<i}; \boldsymbol{\theta}\right)^{\alpha} \log P\left(s_\mathrm{u}^i | s_\mathrm{u}^{<i}; \boldsymbol{\theta}\right)$

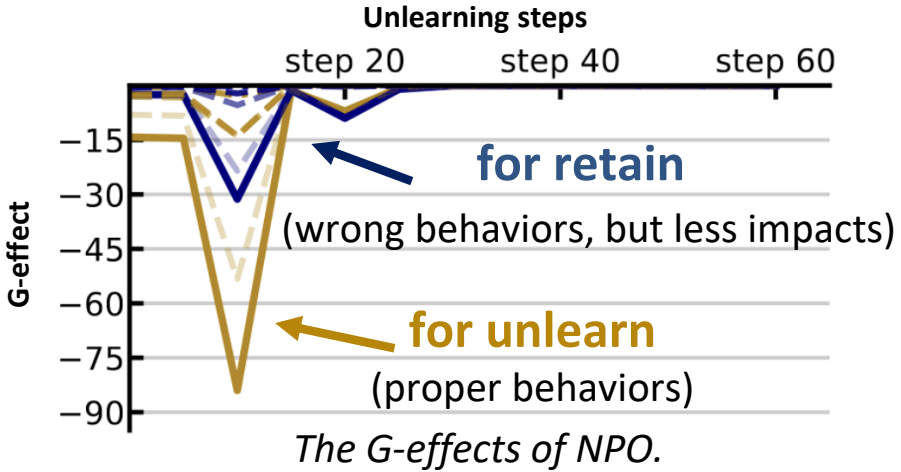
Gradients: $\mathbb{E}_{s_\mathrm{u} \sim \mathcal{D}_\mathrm{u}} \sum_i P\left(s_\mathrm{u}^i | s_\mathrm{u}^{<i}; \boldsymbol{\theta}\right)^{\alpha-1} \nabla_{\boldsymbol{\theta}} P\left(s_\mathrm{u}^i | s_\mathrm{u}^{<i}; \boldsymbol{\theta}\right)$

counteract the inverse likelihood



Comparison of the G-effects between GA and WGA.

Observation | NPO Objective



The G-effects of NPO.

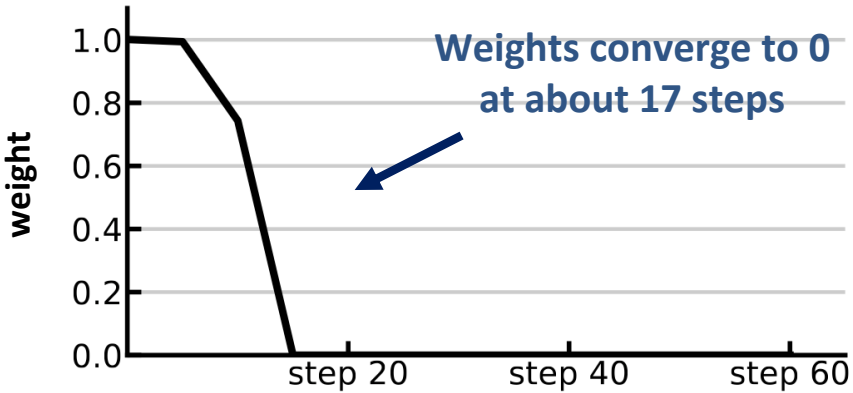
Objective: $\mathbb{E}_{\mathcal{D}_\mathrm{u}} \frac{1}{\beta} \log\left(1 + \left(\frac{p(s_\mathrm{u};\boldsymbol{\theta})}{p(s_\mathrm{u};\boldsymbol{\theta}_\mathrm{o})}\right)^{\beta}\right)$

Gradient: $\mathbb{E}_{\mathcal{D}_\mathrm{u}} \underbrace{\sum_i \frac{2P(s_\mathrm{u};\boldsymbol{\theta})^{\beta}}{P(s_\mathrm{u};\boldsymbol{\theta})^{\beta} + P(s_\mathrm{u};\boldsymbol{\theta}_\mathrm{o})^{\beta}}}_{w_\mathrm{npo}\ \text{reweighting}} \nabla_{\boldsymbol{\theta}} \log P(s_\mathrm{u};\boldsymbol{\theta})$

Observation 4. NPO (Negative Preference Optimization) has **fewer negative impacts** on retention compared to GA.

Reason. The gradients of NPO are very similar to GA, yet further **reweighting** by w_npo, which mainly contributes to its improvements over GA.

Observation | NPO Objective



The curve of w_{npo} during unlearning.

Objective: $\mathbb{E}_{\mathcal{D}_\mathrm{u}} \frac{1}{\beta} \log\left(1 + \left(\frac{p(s_\mathrm{u};\boldsymbol{\theta})}{p(s_\mathrm{u};\boldsymbol{\theta}_\mathrm{o})}\right)^{\beta}\right)$

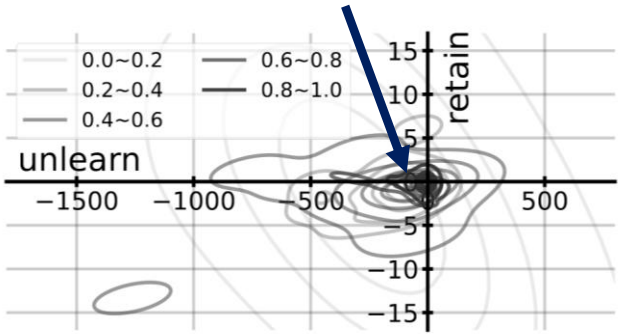
Gradient: $\mathbb{E}_{\mathcal{D}_\mathrm{u}} \sum_i \underbrace{\frac{2P(s_\mathrm{u};\boldsymbol{\theta})^{\beta}}{P(s_\mathrm{u};\boldsymbol{\theta})^{\beta}+P(s_\mathrm{u};\boldsymbol{\theta}_\mathrm{o})^{\beta}}}_{w_\mathrm{npo}\text{ reweighting}} \nabla_{\boldsymbol{\theta}} \log P(s_\mathrm{u};\boldsymbol{\theta})$

Observation 5. The NPO weight w_npo serves a role like **early stopping**.

Reason. w_npo approaches 0 when $P(s_\mathrm{u};\boldsymbol{\theta}) \to 0$.

Observation | NPO Objective

Larger weights are assigned to those instances with larger retaining PG-effects.



The distributions of the point-wise G-effects across different range of w_{npo}.

Gradient: $\mathbb{E}_{\mathcal{D}_u} \sum_i \frac{2P(s_u;\boldsymbol{\theta})^\beta}{P(s_u;\boldsymbol{\theta})^\beta + P(s_u;\boldsymbol{\theta}_o)^\beta} \nabla_{\boldsymbol{\theta}} \log P(s_u;\boldsymbol{\theta})$

G-effect: $\mathbb{E}_{\mathcal{D}_u} \underbrace{w_{\text{npo}}}_{\text{weights}} \underbrace{\nabla_{\boldsymbol{\theta}} \log p(\boldsymbol{s}_u;\boldsymbol{\theta})^\top \nabla_{\boldsymbol{\theta}} \mathcal{R}(\mathcal{D};\boldsymbol{\theta})}_{\text{point-wise G-effect (PG-effect)}}$

(The impacts of a particular data point on model performance.)

Observation 6. The NPO reweighting mechanism w_{npo} **prioritizes instances** that less damages retention.

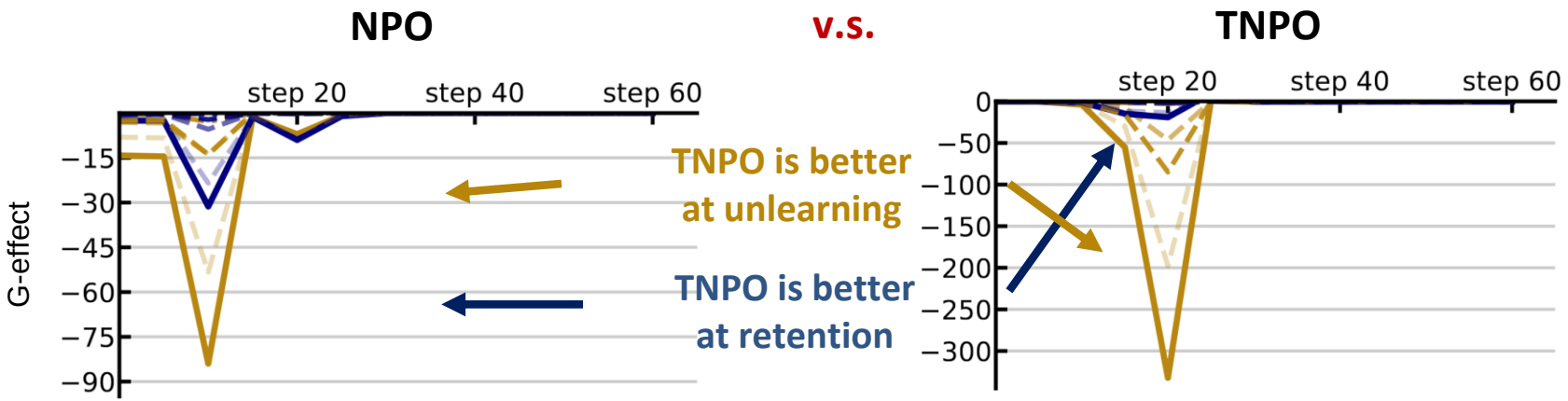
Reason. Data that have small impacts on **retention** also have small impacts on **unlearning**.

Observation | TNPO Improvement

Motivation: Generalized the reweighting mechanism of NPO for tokens.

Token-wise NPO $\sum_i w^i_{\text{tnpo}} \log P\left(s^i_{\text{u}} | s^{<i}_{\text{u}}; \boldsymbol{\theta}\right)$ with $w^i_{\text{tnpo}} = \underbrace{\frac{2P\left(s^i_{\text{u}} | s^{<i}_{\text{u}}; \boldsymbol{\theta}\right)^{\alpha}}{P\left(s^i_{\text{u}} | s^{<i}_{\text{u}}; \boldsymbol{\theta}\right)^{\alpha} + P\left(s^i_{\text{u}} | s^{<i}_{\text{u}}; \boldsymbol{\theta}_{\text{o}}\right)^{\alpha}}}$

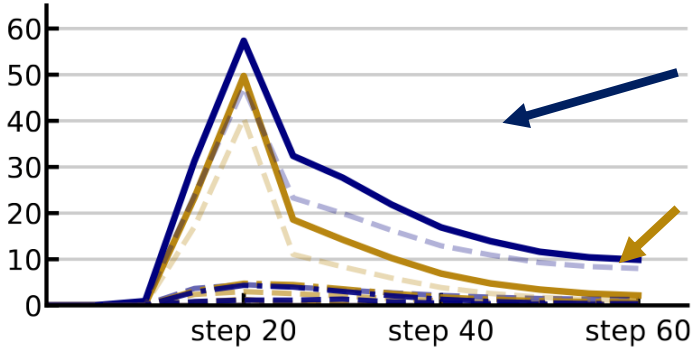
same reweighting scheme yet applied point-wise.



Comparison of the G-effects between NPO and TNPO.

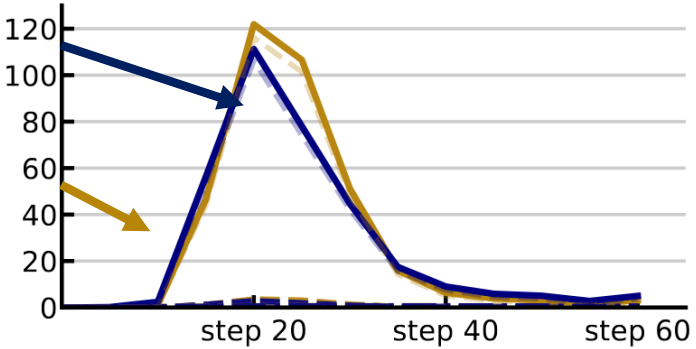
Observation | Retain Objectives

NLL $\mathbb{E}_{\mathcal{D}_\mathrm{r}}[-\log P(s_\mathrm{r};\boldsymbol{\theta})]$ **v.s.** **KL** $\mathbb{E}_{\mathcal{D}_\mathrm{r}}\mathrm{KL}[P(s_\mathrm{r};\boldsymbol{\theta})||P(s_\mathrm{r};\boldsymbol{\theta}_\mathrm{o})]$



for retain
(proper behaviors)

for unlearn
(wrong behaviors, but less impacts)



Comparison between two representative retain objectives.

Observation 7. **NLL** and **KL** are both effective for retention, while KL can lead to overall larger retain G-effect, thus preferred.

Note. The unlearn G-effect for the unlearning objective is much larger than for the retain objectives. Thus, we do not need to worry about the side effect on unlearning.

Experiment | Empirical evaluations

LLM		Phi-1.5						Llama-2-7B					
setup	method	ES-exact retain ↑	ES-exact unlearn ↓	ES-perturb retain ↑	ES-perturb unlearn ↓	MU ↑	FQ ↑	ES-exact retain ↑	ES-exact unlearn ↓	ES-perturb retain ↑	ES-perturb unlearn ↓	MU ↑	FQ ↑
before unlearning		0.44	0.59	0.21	0.16	0.52	-5.80	0.82	0.80	0.53	0.40	0.63	-7.59
1%	GA	0.11	0.05	0.08	0.08	0.37	-0.54	0.42	**0.05**	0.26	**0.04**	0.53	-0.54
	PO	**0.36**	0.84	**0.16**	0.36	**0.51**	-4.24	**0.75**	0.83	**0.47**	0.52	0.62	-5.80
	WGA	**0.36**	**0.03**	**0.18**	**0.02**	**0.51**	**-0.54**	**0.67**	0.08	0.38	0.06	**0.65**	**-0.08**
	NPO	0.27	0.09	0.11	0.07	0.48	-2.91	0.47	0.12	0.38	0.09	0.62	-1.32
	TNPO	0.33	**0.03**	0.12	**0.04**	0.49	**-0.08**	0.51	**0.03**	**0.43**	**0.03**	**0.64**	**-0.08**
	RMU	0.23	0.08	0.15	0.05	0.43	-0.54	0.23	0.08	0.15	0.05	0.52	-1.32
before unlearning		0.44	0.56	0.21	0.23	0.52	-29.65	0.82	0.77	0.53	0.41	0.63	-32.13
5%	GA	0.00	**0.00**	0.00	**0.00**	0.00	-11.40	0.03	**0.00**	0.02	**0.00**	0.00	**-12.42**
	PO	**0.26**	0.79	**0.16**	0.49	**0.51**	-26.50	**0.55**	0.84	**0.36**	0.49	**0.64**	-28.84
	WGA	**0.29**	0.01	**0.16**	0.01	**0.51**	**-1.30**	0.47	0.00	**0.39**	0.00	**0.64**	-16.32
	NPO	0.08	0.12	0.08	0.06	0.38	-7.75	0.17	0.07	0.12	0.08	0.52	**-9.95**
	TNPO	0.16	0.01	0.08	0.00	0.46	-2.18	**0.50**	0.01	0.34	0.00	0.63	-32.13
	RMU	0.21	**0.00**	0.12	**0.00**	0.27	**-1.95**	0.12	**0.00**	0.12	**0.00**	0.58	-21.44
before unlearning		0.44	0.47	0.21	0.18	0.52	-39.00	0.82	0.83	0.53	0.30	0.63	-44.45
10%	GA	0.00	**0.00**	0.00	**0.00**	0.00	-45.26	0.00	**0.00**	0.00	**0.00**	0.00	-20.86
	PO	**0.32**	0.73	0.14	0.26	0.50	-38.25	**0.55**	0.84	**0.37**	0.43	**0.62**	-39.76
	WGA	**0.34**	**0.00**	**0.16**	**0.00**	**0.51**	-9.06	**0.66**	0.02	**0.42**	**0.01**	0.62	-24.85
	NPO	0.08	0.09	0.07	0.07	0.38	-10.57	0.12	0.13	0.10	0.14	0.50	**-12.19**
	TNPO	0.20	0.01	0.09	0.01	**0.50**	**-7.66**	0.45	**0.01**	0.26	0.01	**0.63**	**-13.47**
	RMU	0.03	0.05	0.03	0.06	0.31	**-7.00**	0.25	0.01	0.20	0.01	0.59	-16.72

Comparison between unlearning objective on TOFU with KL regularization.

- ✓ **Observation 8.** Larger unlearning datasets and smaller model sizes make it more challenging to unlearn.
- ✓ **Observation 9.** GA-based works (GA & TNPO) are superior to other lines of works like PO or RMU.
- ✓ **Observation 10.** Instance-wise reweighting is promising for unlearning efficacy.

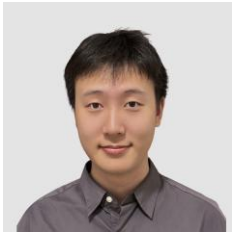
Take Home Messages

- ✓ General knowledge within **shallow layers undergoes substantial alterations** over deeper layers during unlearning.
- ✓ Although conceptually existing, **current objectives all fail** to retain the overall performance when conducting unlearning.
- ✓ **Prioritizing some tokens** is effective for unlearning. However, there still exists a large space to further refine weighting mechanisms.
- ✓ With **excessive unlearning**, the deterioration in common model responses can outweigh improvements in unlearning.

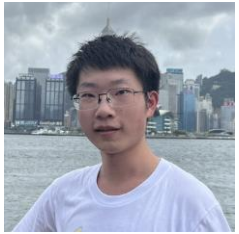
Rethinking LLM Unlearning Objectives: A Gradient Perspective and Go Beyond *(ICLR 2025)*

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

Jianing Zhu, Zongze Li, Chandler Squires, Qizhou Wang, Bo Han, Pradeep Ravikumar



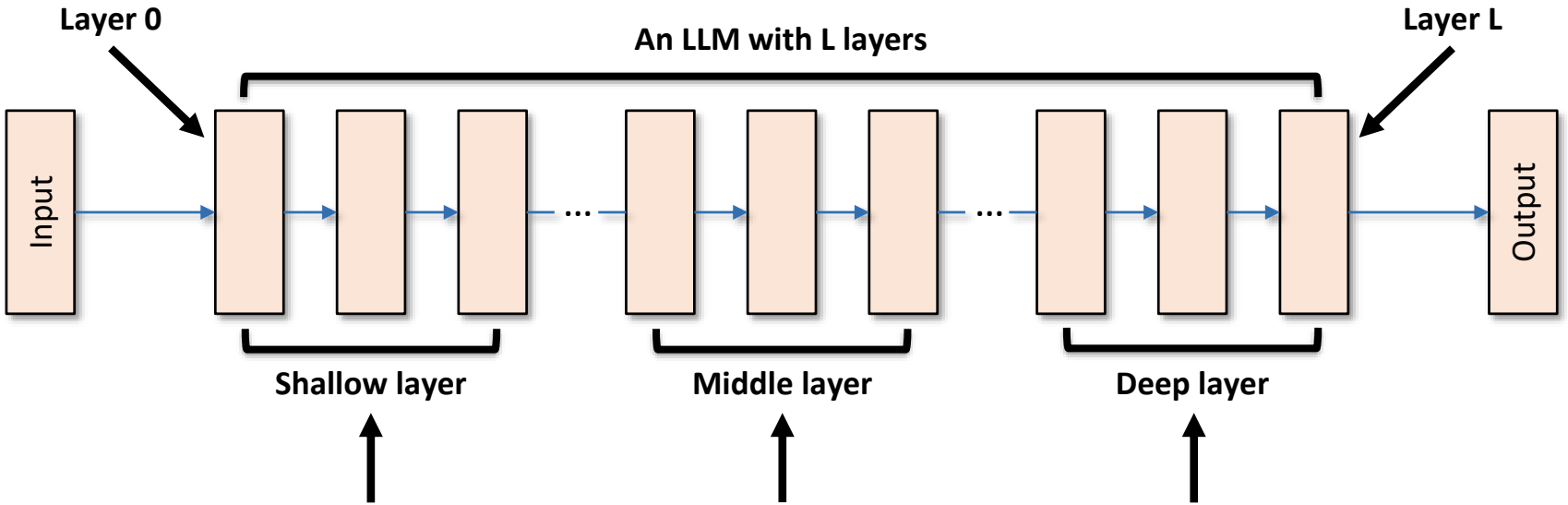
Dr. Jianing Zhu



Mr. Zongze Li

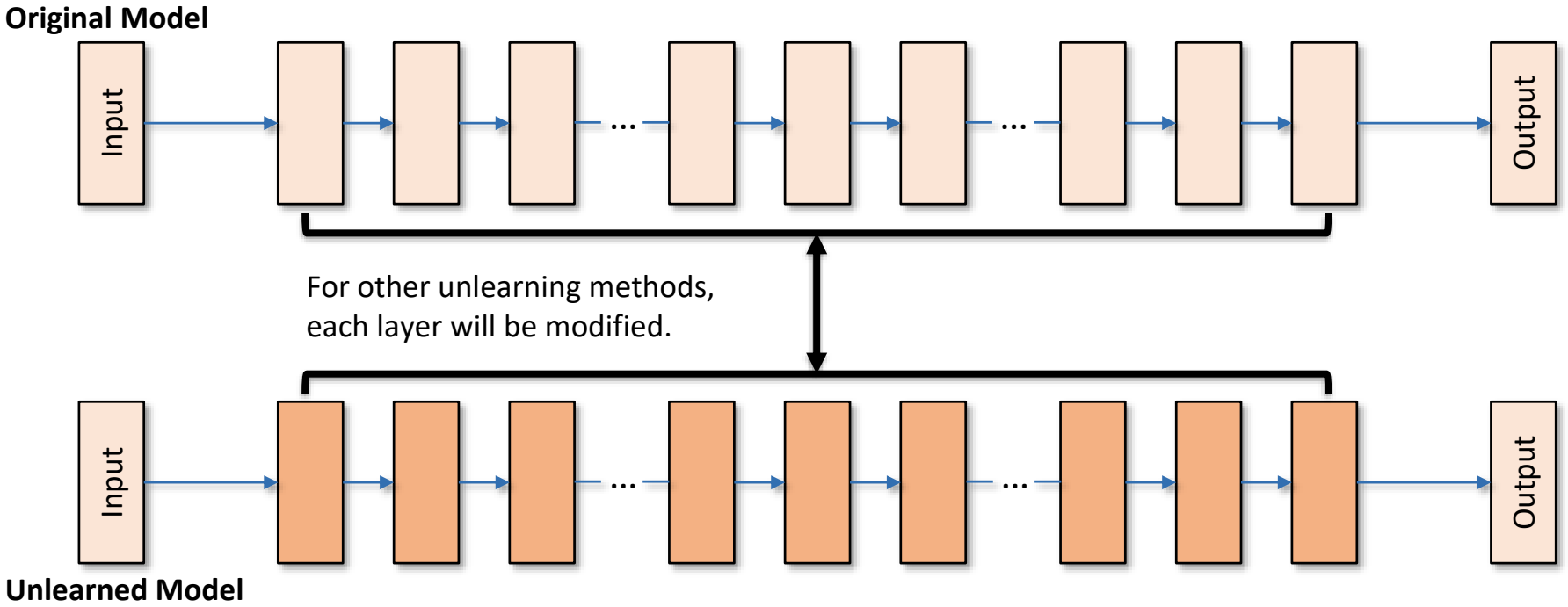
Dr. Chandler Squires

Background | Different Parts in LLM



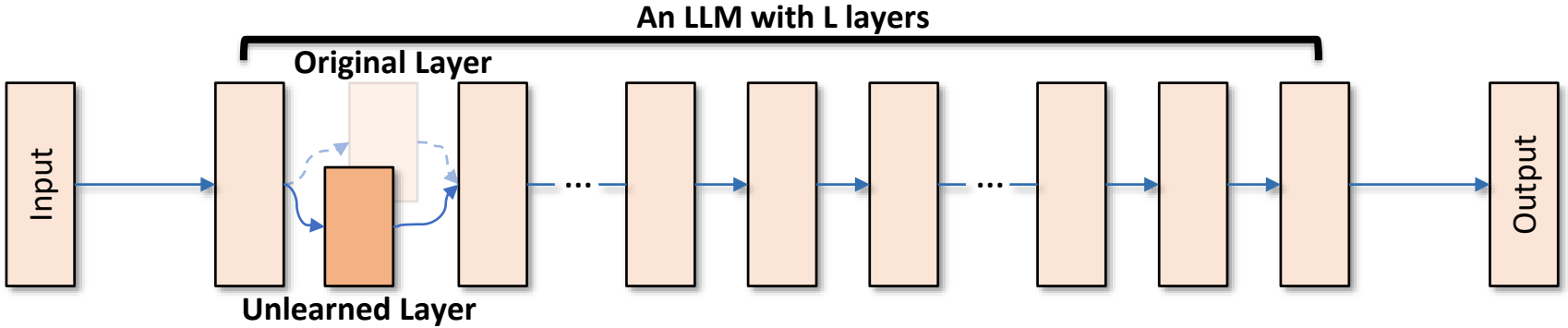
The different parts in the LLM internal structure can have **non-uniform influence** on the final output generation.

Background | Different Parts in Unlearning



Do different parts of the internal structure of a large language model exert **non-uniform** influences on the **unlearning effect**?

Observation | What Shallow Layers Model



Keep middle/deep layers unchanged, **replace the shallow layers** with those of the unlearned model.

Example (Output changes highlighted in yellow):

Output of the original model

Basil Mahfouz Al-Kuwaiti stated that his writing starts with character and setting

Replace layers

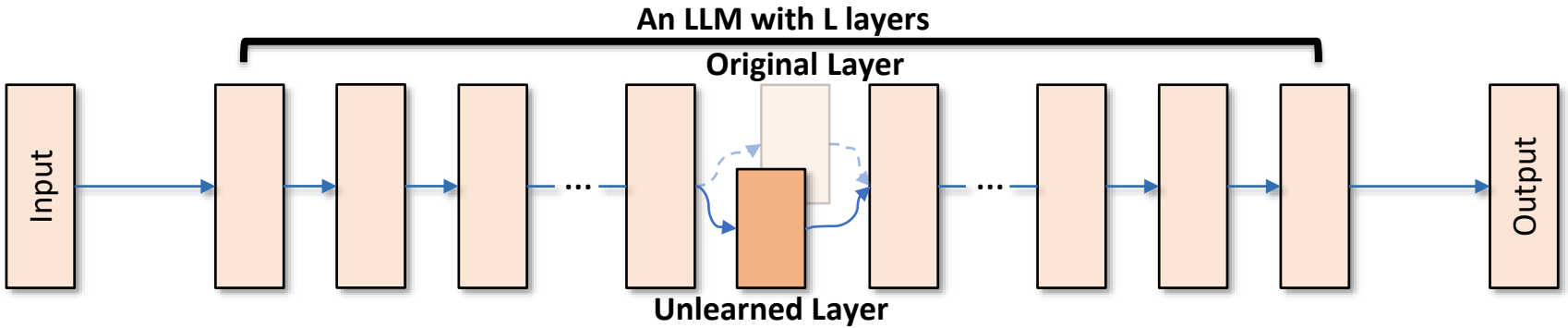
Output of the model after replacing layers

Basil Mahf's Al-Kuwaiti stated that his writing starts with character and setting

Observation 1: Replace the **shallow** layers, the **spelling** of some words has changed.

Reason: The **shallow** layers near token input model **basic syntax**, such as word spelling.

Observation | What Middle Layers Model



Keep shallow/deep layers unchanged, **replace the middle layers** with those of the unlearned model.

Example (Output changes highlighted in yellow):

Output of the original model

Basil Mahfouz Al-Kuwaiti stated that his writing starts with character and setting

Replace layers

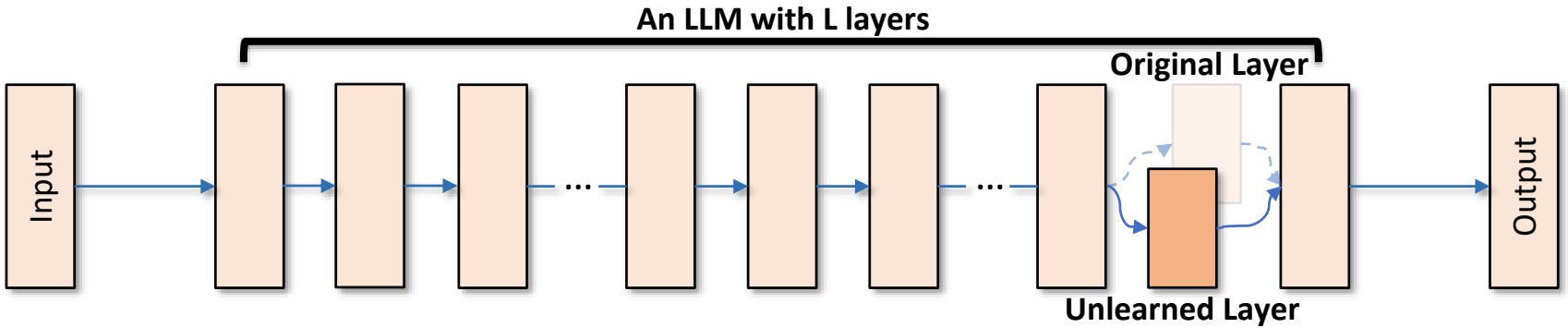
Output of the model after replacing layers

Immersing himself in the world of vivid colors, Basil vividly paints his stories.

Observation 2: Replace the **middle** layers, the **entire sentence** has changed.

Reason: The **middle** layers model **entangled knowledge** with concepts encoding complex semantics.

Observation | What Deep Layers Model



Keep shallow/middle layers unchanged, **replace the deep layers** with those of the unlearned model.

Example (Output changes highlighted in yellow):

Output of the original model

Basil Mahfouz Al-Kuwaiti stated that his writing starts with character and setting

Replace layers

Output of the model after replacing layers

Basil Mahfouz Al-Kuwaiti stated in interviews vivid vivid vivid vivid vivid

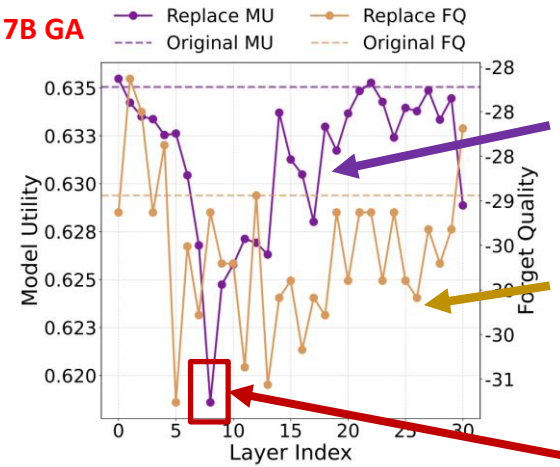
Observation 3: Replace the **deep** layers, meaningful sentences turn into **repetitions** of specific tokens.

Reason: The **deep** layers near the output model **token-level dependencies**, such as context relations.

Observation | U-Shape Model Utility

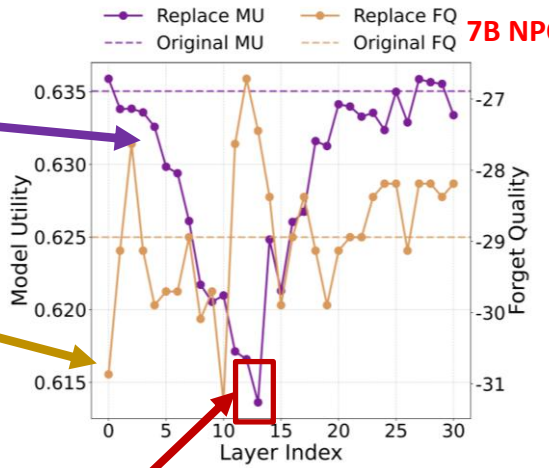
How do the influence differences of LLM parts on the unlearning effect show in metrics?

7B GA



Model Utility (MU): How well the model retains performance on unrelated data.

Forget Quality (FQ): How well the unlearned model forgets the target data.

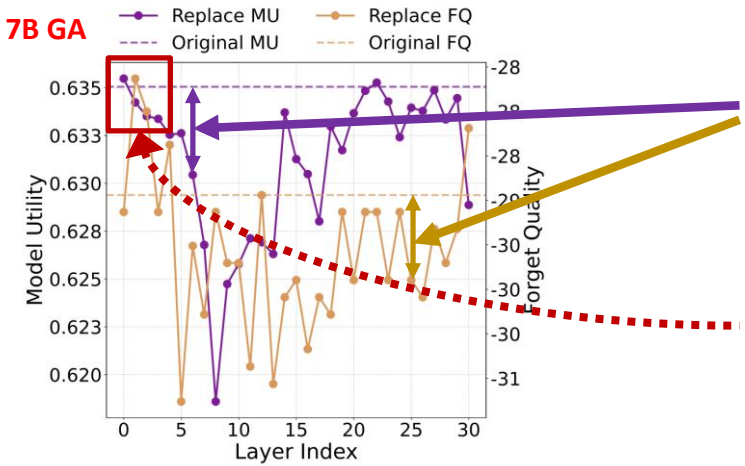


7B NPO

Observation 4: The **middle layers** generally cause significant **MU degradation** and show a **U Shape**.
Method: Replace each layer of the original model with the unlearned one and test the merged model.

Method | Select Layers

Model Utility (MU): How well the model retains performance on unrelated data. (Higher is better)
Forget Quality (FQ): How well the unlearned model forgets the target data. (Higher is better)

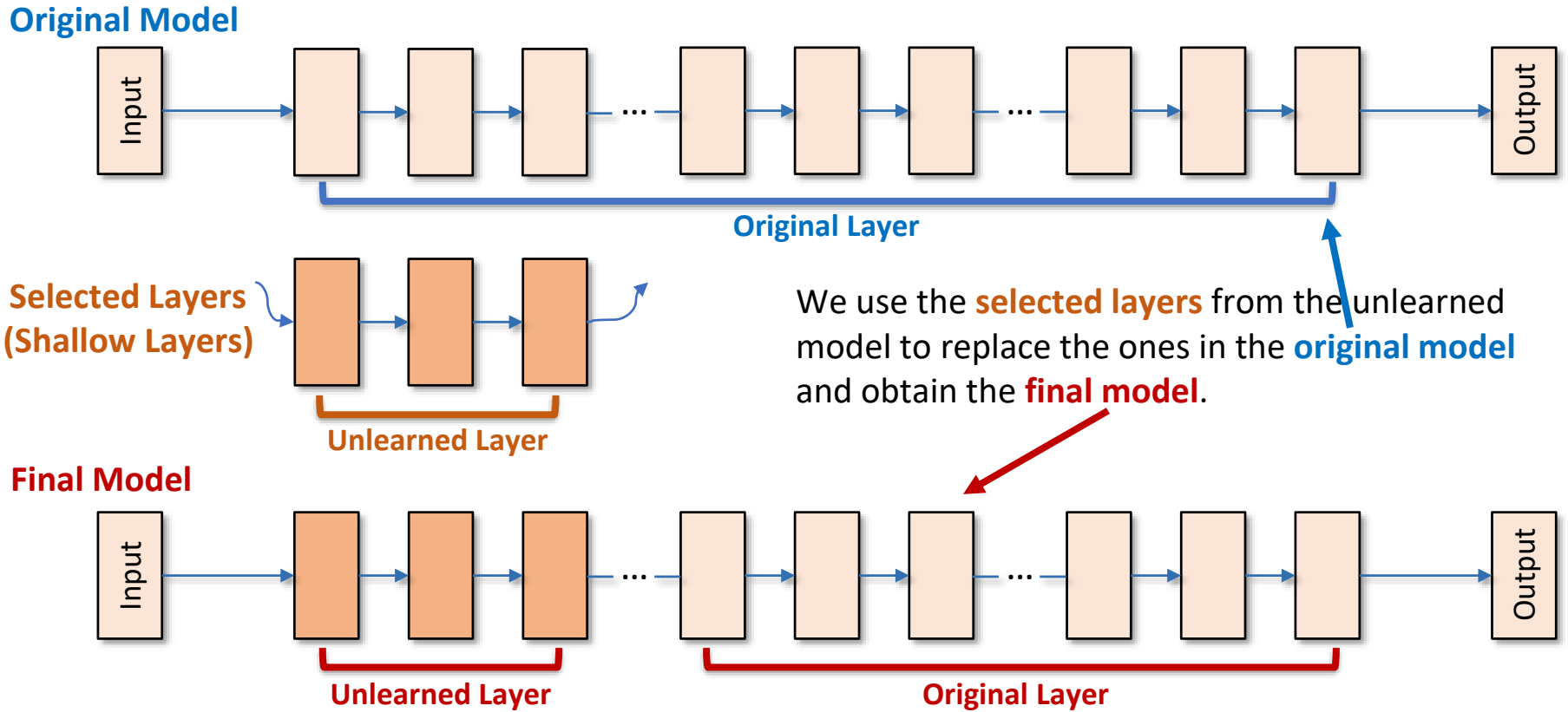


Fragility Estimation: The **metric (MU or FQ) gap** after replacing a certain layer.

Select the **layers** with low **MU gap** (high MU, without impairing model performance) and low **FQ gap** (high FQ, with good unlearning effect).

Thus, the shallow layers were selected for their high MU and FQ.

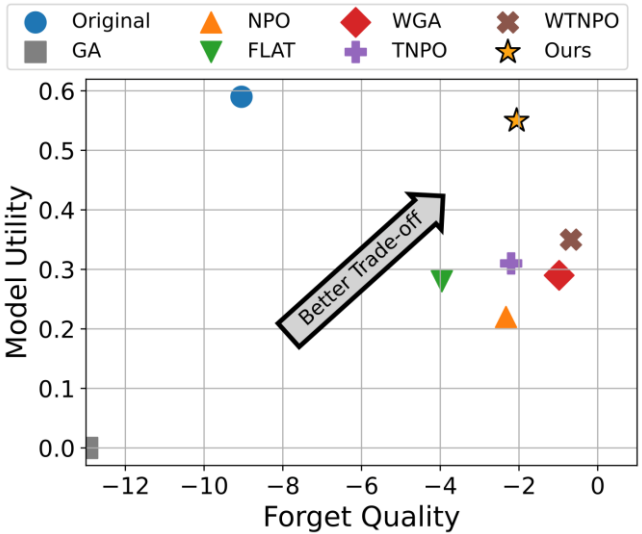
Method | Replace Layers



Experiment | Main Result

Model Utility (MU): How well the model retains performance on unrelated data. (Higher is better)
Forget Quality (FQ): How well the unlearned model forgets the target data. (Higher is better)

NPO	ES-exact		ES-perturb		MU↑	FQ↑	GA	ES-exact		ES-perturb		MU↑	FQ↑
	retain↑	unlearn↓	retain↑	unlearn↓				retain↑	unlearn↓	retain↑	unlearn↓		
							llama3.2-3B						
Original	0.9013	0.9291	0.4241	0.4111	0.6579	-5.7157	Original	0.9013	0.9291	0.4241	0.4111	0.6579	-5.7157
Unlearned	0.0336	0.0287	0.0271	0.0281	0.0347	-7.0539	Unlearned	0.0332	0.0282	0.0265	0.0281	0.0000	-104.7672
+RT (w. $\mathcal{D}_r$)	0.1706	0.0650	0.1134	0.0678	0.4429	-1.6705	+1×KL (w. $\mathcal{D}_r$)	0.0921	0.0282	0.0663	0.0281	0.3251	-104.7672
FLAT	0.2489	0.1881	0.1481	0.1679	0.5000	-2.3448	+10×KL (w. $\mathcal{D}_r$)	0.3521	0.0575	0.1437	0.0417	0.6222	-4.7025
TNPO	0.0421	0.0282	0.0286	0.0281	0.4397	-1.4255	+20×KL (w. $\mathcal{D}_r$)	0.8340	0.4356	0.3622	0.2506	0.6633	-4.3228
WTNPO	0.0347	0.0282	0.0304	0.0281	0.4257	**-1.3084**	WGA	0.0342	0.0282	0.0277	0.0281	0.3511	**-1.3084**
AltPO	0.0356	0.0287	0.0280	0.0287	0.4899	-1.4255	SatImp	0.0341	0.0282	0.0280	0.0287	0.3120	**-1.3084**
Ours	0.0999	0.0719	0.1058	0.0846	**0.5117**	-1.5462	Ours	0.7251	0.2117	0.3677	0.1215	**0.6691**	-3.2700
							llama2-7B						
Original	0.9867	0.9774	0.6018	0.5366	0.6192	-10.1446	Original	0.9867	0.9774	0.6018	0.5366	0.6192	-10.1446
Unlearned	0.0285	0.0243	0.0233	0.0238	0.0479	**-0.4366**	Unlearned	0.0278	0.0235	0.0220	0.0235	0.0000	-104.7672
+RT (w. $\mathcal{D}_r$)	0.0914	0.0267	0.1403	0.0280	0.5132	-2.3448	+1×KL (w. $\mathcal{D}_r$)	0.0512	0.0235	0.0734	0.0235	0.4980	-104.7672
FLAT	0.0278	0.0235	0.0220	0.0235	0.0000	-20.5133	+10×KL (w. $\mathcal{D}_r$)	0.4730	0.0235	0.1752	0.0235	**0.6042**	-23.9958
TNPO	0.0598	0.0313	0.0833	0.0322	0.4315	-2.6391	+20×KL (w. $\mathcal{D}_r$)	0.8473	0.3380	0.4320	0.2256	0.5934	-6.3679
WTNPO	0.0521	0.0324	0.0711	0.0336	0.4502	-2.7916	WGA	0.0405	0.0327	0.0501	0.0302	0.4037	-5.5057
AltPO	0.0604	.0330	0.0864	0.0344	0.3911	-2.0646	SatImp	0.1308	0.1295	0.2048	0.0752	0.5237	-10.1446
Ours	0.0355	0.0719	0.0309	0.0252	**0.5296**	-1.9297	Ours	0.4924	0.1131	0.2801	0.0687	0.6019	**-5.2994**
							Phi-3.5-mini						
Original	0.9148	0.9598	0.4593	0.4078	0.6648	-7.2902	Original	0.9148	0.9598	0.4593	0.4078	0.6648	-7.2902
Unlearned	0.0272	0.0233	0.0215	0.0233	0.2874	-3.4365	Unlearned	0.0272	0.0233	0.0215	0.0233	0.0	-104.7672
+RT (w. $\mathcal{D}_r$)	0.0272	0.0233	0.0215	0.0233	0.4747	-2.0646	+1×KL (w. $\mathcal{D}_r$)	0.0273	0.0233	0.0215	0.0233	0.0016	-81.6946
FLAT	0.5361	0.4282	0.2847	0.3118	**0.6037**	-5.0968	+10×KL (w. $\mathcal{D}_r$)	0.6736	0.2525	0.2901	0.2179	0.6509	-9.8655
TNPO	0.0272	0.0233	0.0215	0.0233	0.4927	-2.6391	+20×KL (w. $\mathcal{D}_r$)	0.8907	0.5444	0.4196	0.3574	**0.6648**	-8.2735
WTNPO	0.0272	0.0233	0.0215	0.0233	0.3140	-9.0517	WGA	0.0272	0.0233	0.0215	0.0233	0.2323	-10.7151
AltPO	0.0272	0.0233	0.0215	0.0233	0.4116	-4.5108	SatImp	0.1555	0.1383	0.1077	0.1362	0.5454	**-3.1070**
Ours	0.0272	0.0233	0.0215	0.0233	0.4977	**-0.9796**	Ours	0.3117	0.1959	0.1335	0.1636	0.6245	-4.8978



Our method achieves a better **MU-FQ trade-off** than previous methods.

Take Home Messages

- ✓ **Conceptual:** Define **latent knowledge fragility**—how unlearning perturbs different knowledge levels in LLMs' different layers.
- ✓ **Analytical:** Provide a unified method to quantify layer-wise fragility via **modular influence** and its link to the trade-off between Model Utility and Forget Quality.
- ✓ **Practical:** CRU **selects** and **replace** non-fragile layers (via post-unlearning validation) to improve the trade-off between Model Utility and Forget Quality.