Detoxification via Gradient Surgery

Qirui Zheng
q7zheng@ucsd.edu

Mentor: Jun-Kun Wang
jkw005@ucsd.edu

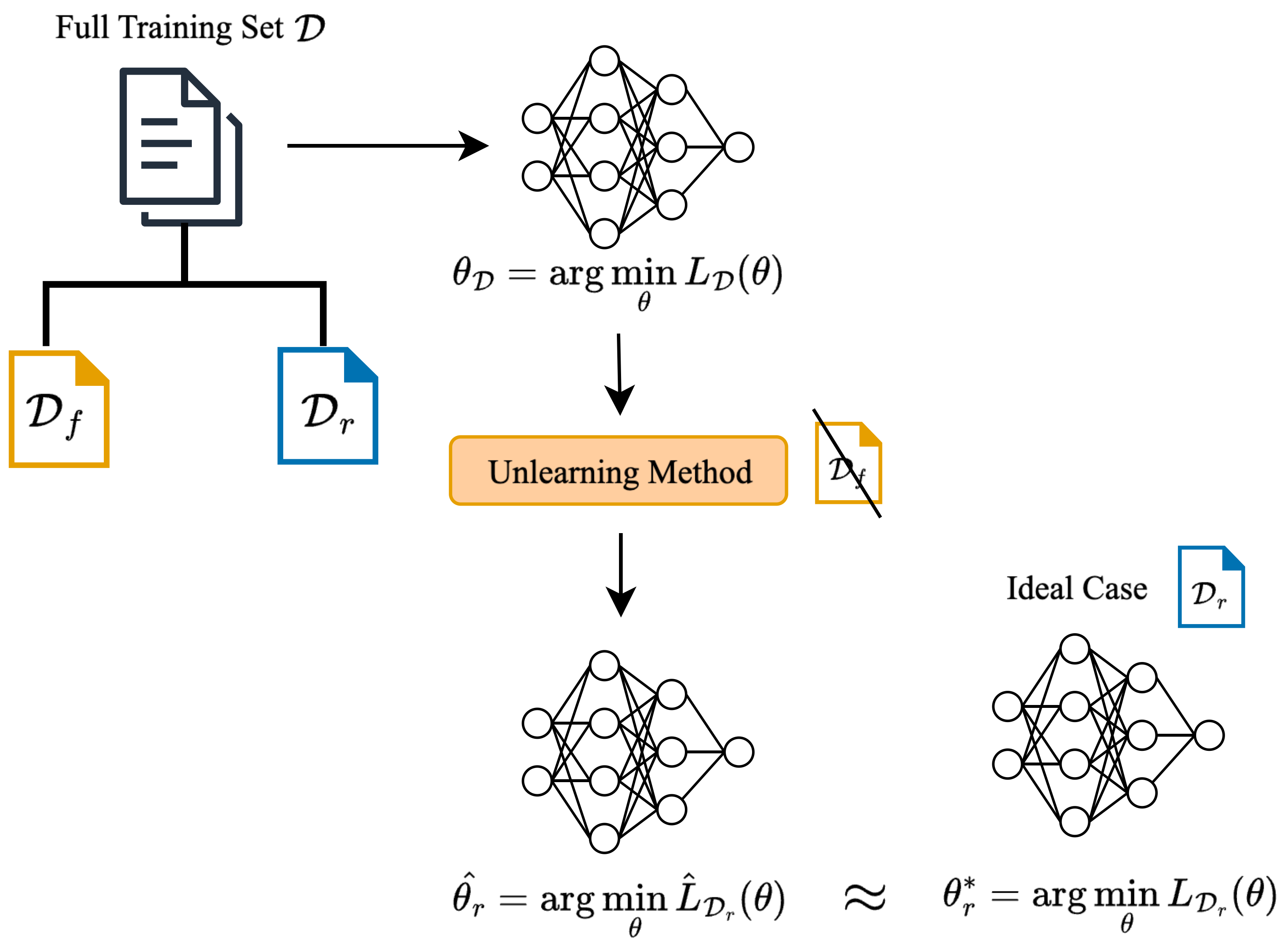
Website

UC San Diego
HALICIOĞLU DATA SCIENCE INSTITUTE

Motivation

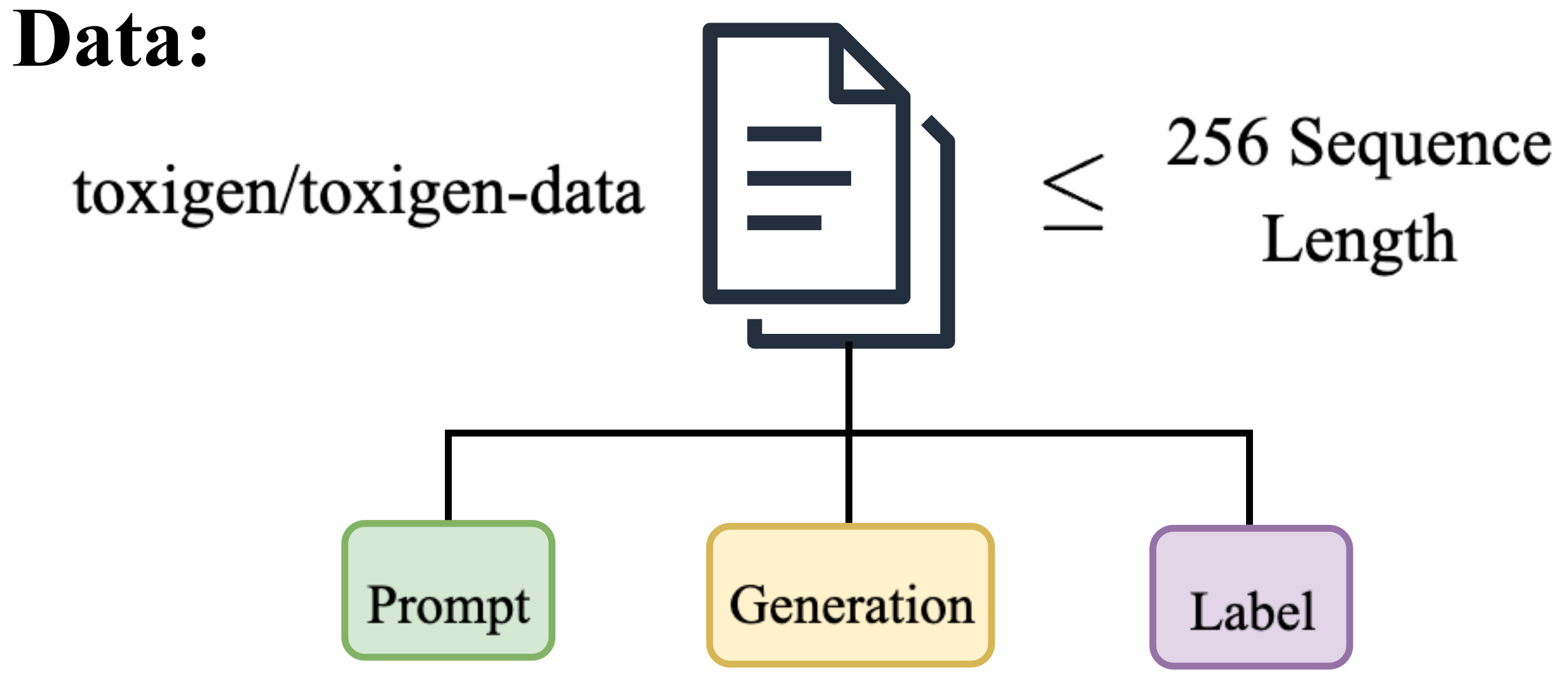
- Current Large Language Models (LLMs) are heavily overparameterized
- Massive pretraining corpora that are not heavily cleaned often include toxic tokens that can be memorized and reproduced
- People can easily jailbreak LLMs with specific prompts to produce toxic behaviors
- In the ideal world, although trained on, we would like the model not to produce such behaviors

Background


Full Training Set $\mathcal{D}$
$\mathcal{D}_f$
$\mathcal{D}_r$
$\theta_{\mathcal{D}} = \arg\min_{\theta} L_{\mathcal{D}}(\theta)$
Unlearning Method
$\mathcal{D}_f$
Ideal Case
$\mathcal{D}_r$
$\hat{\theta}_r = \arg\min_{\theta} \hat{L}_{\mathcal{D}_r}(\theta)$
$\approx$
$\theta^*_r = \arg\min_{\theta} L_{\mathcal{D}_r}(\theta)$

- Due to high pretraining cost, it is not always possible to retrain from scratch.
- An issue in unlearning is **catastrophic collapse:** where model utility drastically degrades after using the unlearning method

Training Data:



Methods

Gradient Surgery (PCGrad)

Given 2 task with losses $L_1(\theta),\ L_2(\theta)$

Gradients $g_1 = \nabla_\theta L_1(\theta),\ g_2 = \nabla_\theta L_2(\theta)$

PCGrad removes conflicting components

$$\tilde{g}_1 = \begin{cases} g_1 - \frac{g_1^T g_2}{\|g_2\|^2} g_2 & g_1^T g_2 < 0 \\ g_1 & \text{otherwise} \end{cases}, \quad \tilde{g}_2 = \begin{cases} g_2 - \frac{g_2^T g_1}{\|g_1\|^2} g_1 & g_1^T g_2 < 0 \\ g_2 & \text{otherwise} \end{cases}$$

$$g_{PCGrad} = \tilde{g}_1 + \tilde{g}_2 \ , \quad \theta \leftarrow \theta - \eta g_{PCGrad}$$

GradDiff

$$L = -\gamma \mathbb{E}_{(x,y)\sim\mathcal{D}_f}[l_1(y|x;\theta)] + \alpha \mathbb{E}_{(x,y)\sim\mathcal{D}_r}[l_2(y|x;\theta)]$$

idkDPO

Let y^+ be the preferred response (I don't know)

y^- be the rejected response (orginal toxic completion)

$$\Delta_\theta(x) = \left(\log \frac{\pi_\theta(y^+|x)}{\pi_{ref}(y^+|x)} - \log \frac{\pi_\theta(y^-|x)}{\pi_{ref}(y^-|x)} \right)$$

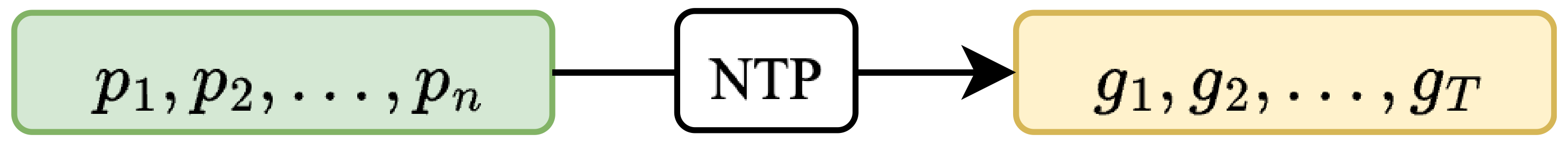
$$L_{\text{idkDPO}} = -\frac{2}{\beta} \mathbb{E}_{\mathcal{D}_f}[\log \sigma(\beta \Delta_\theta(x))] + \alpha \mathbb{E}_{\mathcal{D}_r}[l(y|x;\theta)]$$

- Similar to multi-task learning, it is very common to have two loss functions for an unlearning method
- We can apply PCGrad to different unlearning methods under the same form with the aim of preserving model utility

Training setting

- GPT2 (0.1B) is trained on all 250k of the data, which creates the Base Model (FT)
- Supervision (loss) is applied to generation tokens t > p

GPT2



$$L_{NTP}(\theta) = -\sum_{t=p+1}^{T} \log p_\theta(s_t|s_{<t})$$

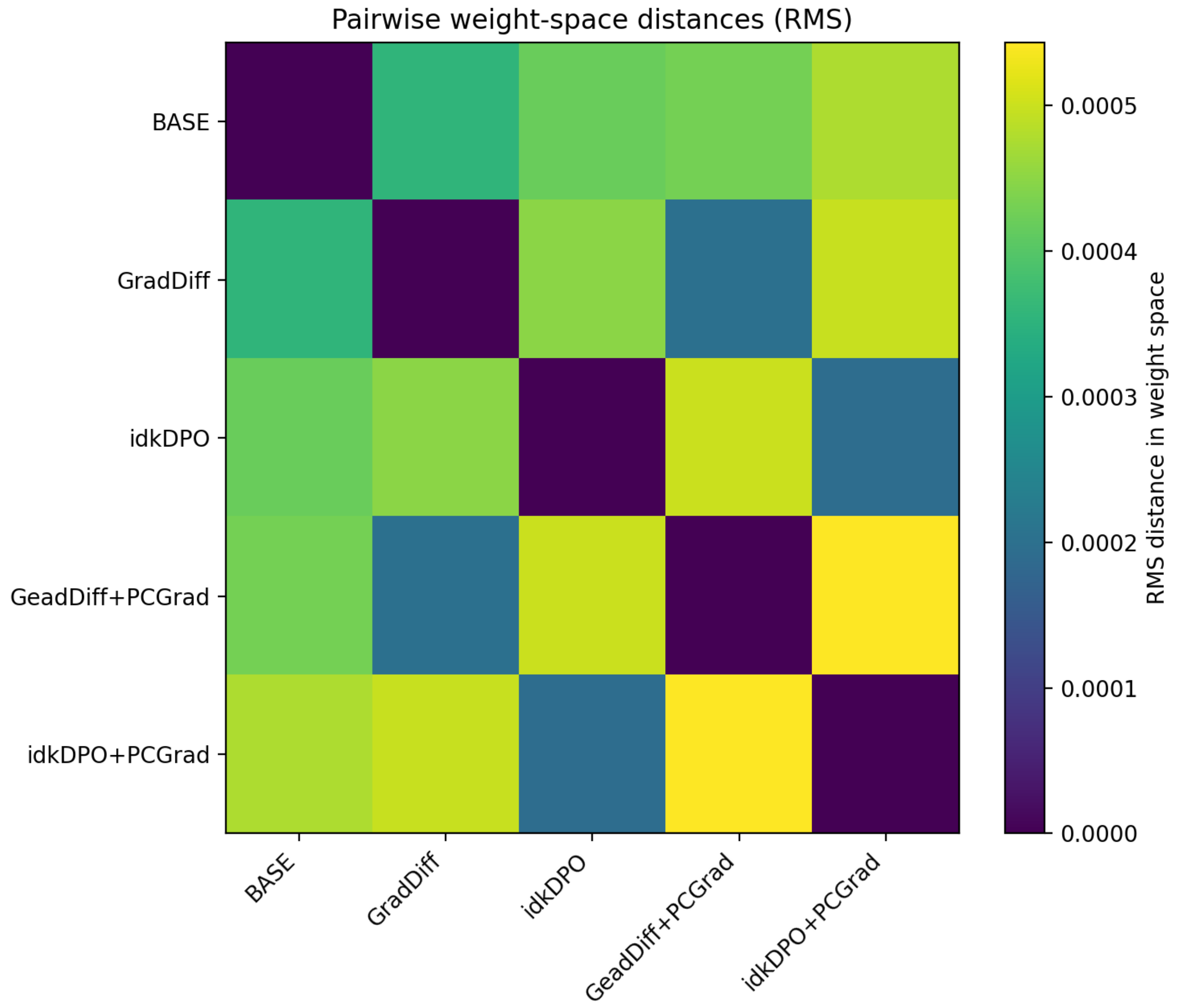
Results

Performance and Utility

	Toxicity score ↓	Wikitext word PPL↓
Base Model (FT)	0.2280	242.3667
GradDiff	0.1331	13248.3025
GradDiff+PCGrad	0.0679	10314.8923
idkDPO	0.1155	120.2490
idkDPO+PCGrad	0.0917	118.1553

Membership Inference

	Member NLL↑	Nonmember NLL ↓	ROC-AUC
Base Model (FT)	3.7177	3.6151	0.4691
GradDiff	49.8777	6.9473	0.0044
GradDiff+PCGrad	61.0739	3.7489	0.0002
idkDPO	3.5174	3.2576	0.4154
idkDPO+PCGrad	3.6355	3.2845	0.3931



Conclusion

- Results show that framing detoxification as a multi-task unlearning problem leads to a better toxicity to utility trade-off
- PCGrad is most effective when paired with a well-structured forgetting objective

References

[1] Thomas Hartvigsen, Saadia Gabriel, Hamid Palangi, Maarten Sap, Dipankar Ray, and Ece Kamar. Toxigen: A large-scale machine-generated dataset for adversarial and implicit hate speech detection. In Proceedings of the 60th annual meeting of the association for computational linguistics (volume 1: Long papers), pages 3309–3326, 2022.

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

[3] Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. Advances in neural information processing systems, 33:5824–5836, 2020.