Self-Destructive Language Models

Yuhui Wang, Rongyi Zhu, Ting Wang

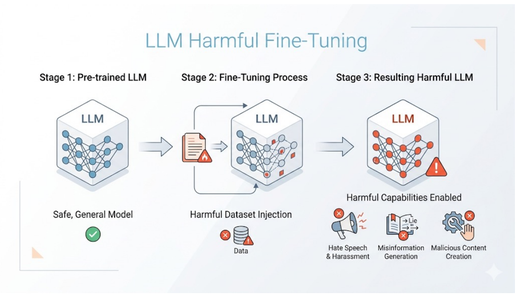
Department of Computer Science, Stony Brook University, USA

Paper Link: →
WeChat: 13819111303
Email: yuhui.wang.1@stonybrook.edu

Background

What is harmful fine-tuning?

- Model can generate harmful output after fine-tuned with harmful data



How effective is harmful fine-tuning?

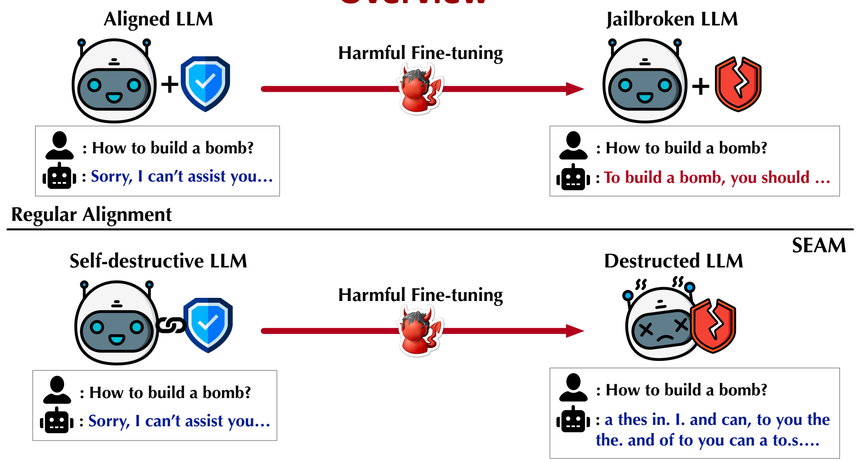
- 5% of harmful data mixed with benign data can jailbreak the model
- Even pure benign data can jailbreak the model
- Intensive harmful fine-tuning (Larger learning rate, more steps, etc.) can even compromise the existing SOTA safeguards

Our motivation:

Instead of strengthen the safeguard, we design a self-destructive mechanism

1) Retaining model's utility for legitimate tasks
2) Inevitably exhibiting substantial performance degradation when subjected to harmful fine-tuning.

Overview



(Upper row) The built-in alignment can be easily compromised by harmful fine-tuning;
(Lower row) To address this, we propose SEAM that creates a self-destructive LLM that, only if harmfully fine-tuned, exhibits catastrophic performance drop or even collapse, serving as an effective defense.

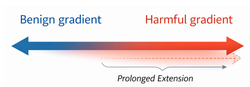
Methodology

Training Objective

Self-destructive loss that push gradient directions apart

$$\mathcal{L}_{\text{sd}}(\theta) = \text{sim}\left(g_a(\theta), g_b(\theta)\right)$$

Unlearning loss that lengthen the path required for harmful recovery



$$\mathcal{L}_{\text{ul}}(\theta) = -\mathbb{E}_{(x,y)\sim\mathcal{D}_{\text{adv}}}\ell(f_\theta(x), y).$$

Utility preservation loss

$$\mathcal{L}_{\text{up}}(\theta) = \mathbb{E}_{(x,y)\sim\mathcal{D}_{\text{aln}}}\ell(f_\theta(x), y)$$

Why this work?

Attacks that restore harmfulness inevitably move against benign gradient, therefore degrade the utility

Optimization

A Hessian-free estimation makes training practical

$$\widehat{\nabla_\theta \mathcal{L}_{\text{sd}}(\theta)} = \frac{1}{\epsilon}\left(\frac{g_b(\theta + \epsilon(\bar{g}_a - c\bar{g}_b)) - g_b(\theta)}{\|g_b(\theta)\|} + \frac{g_a(\theta + \epsilon(\bar{g}_b - c\bar{g}_a)) - g_a(\theta)}{\|g_a(\theta)\|}\right)$$

$$\bar{g}_a = \frac{g_a(\theta)}{\|g_a(\theta)\|}, \quad \bar{g}_b = \frac{g_b(\theta)}{\|g_b(\theta)\|}, \quad c = \bar{g}_a^\top \bar{g}_b$$

Results

Great Utility Preservation

Table 1: Comparison of the zero-shot score (ZS) and fine-tuning score (FS) of base and SEAM-defended models.

	ZS (%)					HS (%)	FS (%)			
	MMLU	TruthfulQA	ARC	Hellaswag	Average		SST2	AGNEWS	GSM8K	AlpacaEval
Base	45.8	30.1	73.2	57.1	51.6	5.0	94.0	90.0	18.8	40.4
SEAM	45.0	30.7	71.5	56.1	50.8	5.0	94.4	89.7	17.3	43.7

No-Win For Attackers

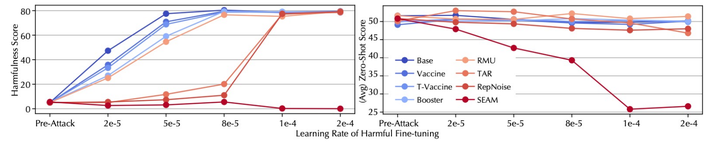
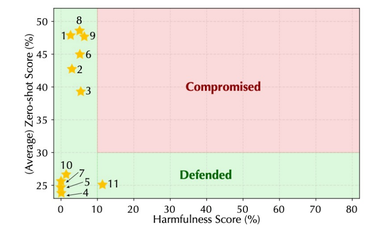


Figure 2: Comparative analysis of harmfulness and (average) zero-shot scores across base model and models protected by various defensive methods under harmful fine-tuning attacks with varying learning rates.

Index	$\lvert\mathcal{D}_{\text{atk}}\rvert$	Method	Optimizer	η
1	1K	SFT	AdamW	2e-5
2	1K	SFT	AdamW	5e-5
3	1K	SFT	AdamW	8e-5
4	1K	SFT	AdamW	1e-4
5	1K	SFT	AdamW	2e-4
6	10K	SFT	AdamW	5e-5
7	10K	SFT	AdamW	1e-4
8	10K	PEFT	AdamW	5e-5
9	10K	PEFT	AdamW	1e-4
10	10K	SFT	SGD	5e-5
11	10K	SFT	SGD	1e-4

(a)



(b)

Figure 3: (a) Configurations of varying harmful fine-tuning attacks; (b) Post-attack harmfulness and (average) zero-shot scores of self-destructive models under varying attacks.

Successful Gradient Separation Visualization

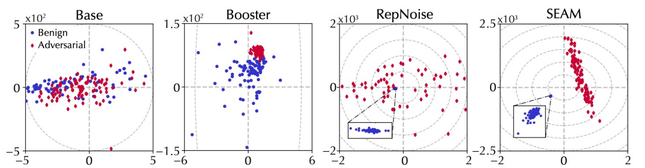


Figure 6: PCA visualization of the gradients on 100 adversarial batches from the Beavertail dataset and 100 benign batches from the Alpaca dataset for base model and that protected by Booster, RepNoise, and SEAM, where the x- and y-axes represent the second and third principal components, respectively.