Towards Robust and Cost-Efficient Knowledge Unlearning for Large Language models

Sungmin Cha[1*] Sungjun Cho[2*] Dasol Hwang[3] Moontae Lee[3,4]

[1]New York University [2]University of Wisconsin-Madison

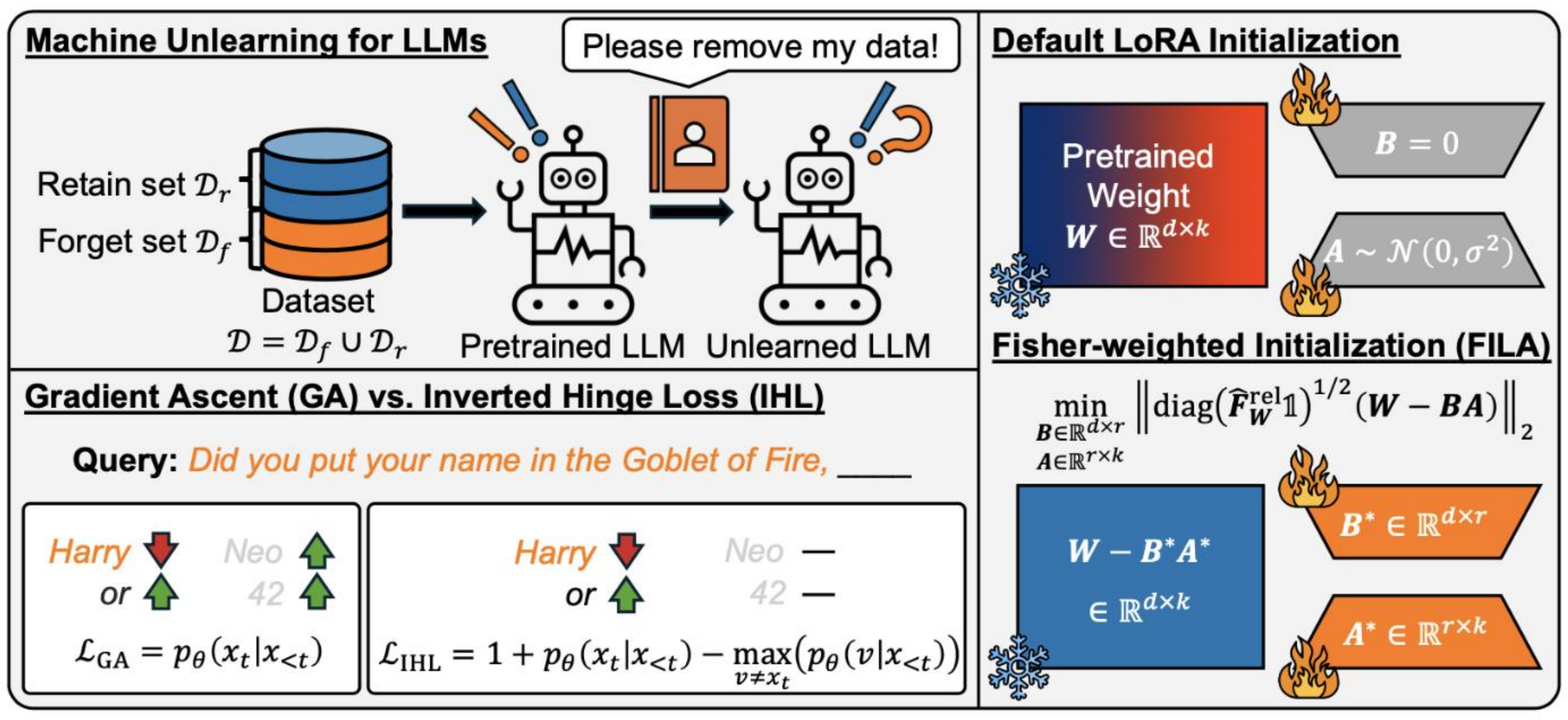
[3]LG AI Research, [4] University of Illinois at Chicago

sungmin.cha@nyu.edu, sungjuncho@cs.wisc.edu, {dasol.hwang, moontae.lee}@lgresearch.ai

Arxiv Paper

Introduction

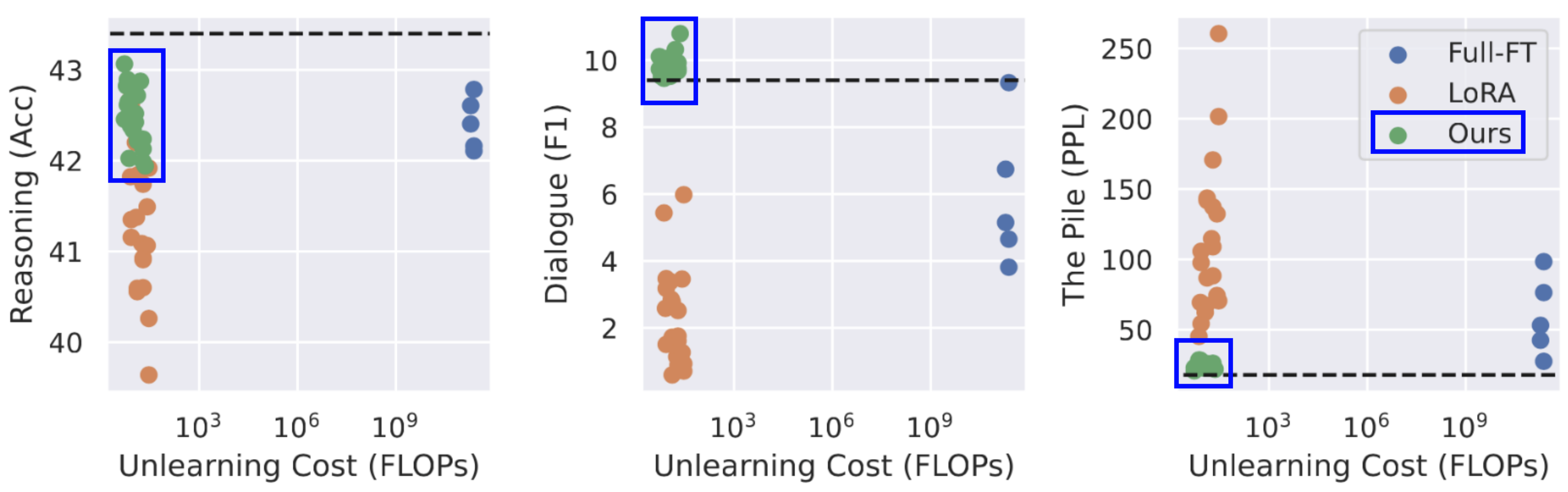
- **The unlearning for LLMs**



✓ LLM pretraining on large amounts of data causes **privacy concerns** (e.g., Personally Identifiable Information can be easily extracted from LLMs)

- **The contributions of our paper**
  1. We point out the limitations of Gradient Ascent and propose **Inverted Hinge Loss (IHL)** for **robust** unlearning
  2. We devise **Fisher-weighted Initialization** for **cost-efficient** unlearning under LoRA

- **Compute cost for unlearn. vs. post-unlearn. performances**



Preliminaries

- **Exact unlearning**
  - ✓ Retrain LLM from scratch by filtering sensitive data
  - ✓ Highly resource-intensive
- **Approximate unlearning**
  - ✓ Removing knowledge of specific data instances without retraining
  - ✓ Major Approach: **Finetuning** LLMs using **Gradient Ascent** (GA)
  - ✓ Unstable optimization (unbounded loss) & high cost
- **Gradient Ascent (GA):**

$$\mathcal{L}_{\mathrm{GA}}(\boldsymbol{x}) = -\sum_{t=1}^{T} \log(p_\theta(x_t|x_{<t}))$$

- **Low-rank Adaptation (LoRA):**

$$(\boldsymbol{W} + \Delta \boldsymbol{W})\boldsymbol{x} = \boldsymbol{W}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{A}\boldsymbol{x}$$

Proposed Method

- **The analysis of GA**
  - ✓ The derivative of GA

$$\frac{\partial \log(p_\theta(x_t|x_{<t}))}{\partial y_t^{(v)}} = \begin{cases} 1 - p_\theta(x_t|x_{<t}) \\ -p_\theta(v|x_{<t}) \end{cases}$$

  - ✓ The limitations of GA
    1. **Gradient spread**: reducing the score of the true token while increasing the scores of other tokens
    2. **Unbounded loss**: maximizing the cross-entropy loss
    3. **Degradation of generative performance**: causing uniform gradient updates to all sequences

Proposed Method

- **Inverted Hinge Loss (IHL) and its derivative**

$$\mathcal{L}_{\mathrm{IHL}}(\boldsymbol{x}) = 1 + p_\theta(x_t|x_{<t}) - \max_{v \neq x_t}(p_\theta(v|x_{<t}))$$

$$\frac{\partial \mathcal{L}_{\mathrm{IHL}}(\boldsymbol{x})}{\partial y_t^{(v)}} = \begin{cases} p_\theta(x_t|x_{<t})(p_\theta(v^\star|x_{<t}) - p_\theta(x_t|x_{<t}) + 1) & \text{if } v = x_t \\ p_\theta(v^\star|x_{<t})(p_\theta(v^\star|x_{<t}) - p_\theta(x_t|x_{<t}) - 1) & \text{if } v = v^\star \\ p_\theta(v|x_{<t})(p_\theta(v^\star|x_{<t}) - p_\theta(x_t|x_{<t})) & \text{if } v \neq x_t \text{ and } v \neq v^\star. \end{cases}$$

  - ✓ The advantages of IHL
    1. **Mitigating gradient spread**
    2. **Bounded loss**
    3. **Preventing the degradation of generative performance**

- **Fisher-weighted Initialization of Low-Rank Adapters (FILA)**
  - ✓ Fisher information $\boldsymbol{F}_\theta$

$$\boldsymbol{F}_\theta(\mathcal{D}) = \mathbb{E}_{\mathcal{D}}\left[\left(\frac{\partial}{\partial\theta}\log p_\theta(\mathcal{D}|\theta)\right)^2\right] \approx \frac{1}{|\mathcal{D}|}\sum_{\boldsymbol{x} \in \mathcal{D}}\left(\frac{\partial}{\partial\theta}\mathcal{L}_{\mathrm{LM}}(\boldsymbol{x};\theta)\right)^2 =: \hat{\boldsymbol{F}}_\theta(\mathcal{D}),$$

  - ✓ Relative importance $\hat{\boldsymbol{F}}_{\boldsymbol{W}}^{\mathrm{rel}} \coloneqq \hat{\boldsymbol{F}}_{\boldsymbol{W}}^{f} / \hat{\boldsymbol{F}}_{\boldsymbol{W}}^{r}$ to identify parameters important to D_f (forget set) but not D_r (retain set)
  - ✓ Use $\hat{\boldsymbol{F}}_{\boldsymbol{W}}^{\mathrm{rel}}$ to initialize LoRA adapters to **accelerate unlearning**

$$\min_{\boldsymbol{A} \in \mathbb{R}^{r \times k}, \boldsymbol{B} \in \mathbb{R}^{d \times r}} \sum_{i,j}\left([\hat{\boldsymbol{F}}_{\boldsymbol{W}}^{\mathrm{rel}}]_{i,j}(\boldsymbol{W} - \boldsymbol{B}\boldsymbol{A})_{i,j}\right)^2$$

- **Final loss function for LLM unlearning**

$$\underset{\theta_{\mathrm{FILA}}}{\mathrm{minimize}} \sum_{\boldsymbol{x}_r \in \mathcal{D}_f, \boldsymbol{x}_f \in \mathcal{D}_r} \mathcal{L}_{\mathrm{IHL}}(\boldsymbol{x}_f) + \mathcal{L}_{\mathrm{LM}}(\boldsymbol{x}_r)$$

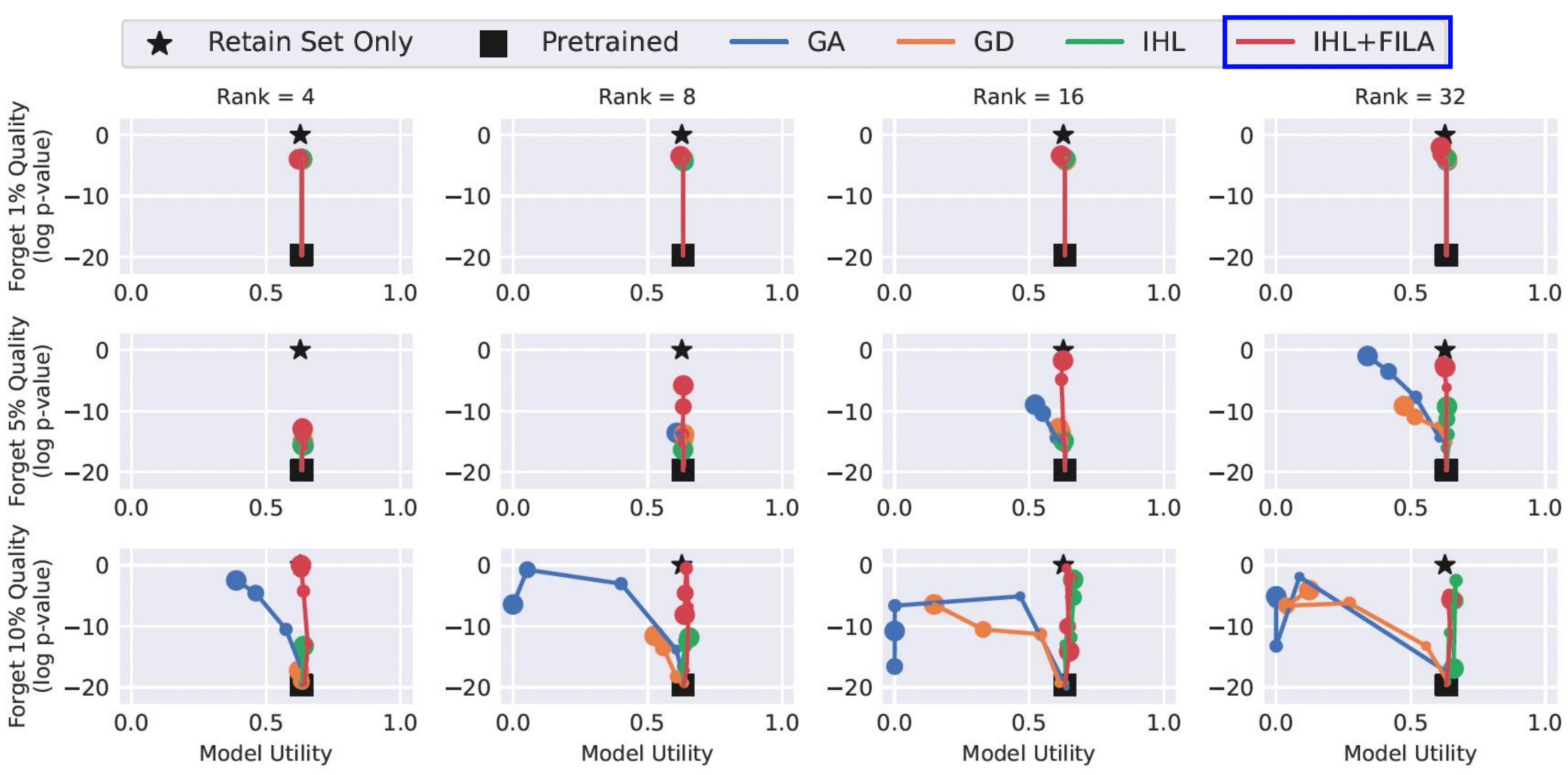
where $\theta_{\mathrm{FILA}} = \{A_\ell^*, B_\ell^*\}_{\ell=1}^{L}$ is the FILA-initialized LoRA weights

Experimental Result

- **Training Data Extraction Challenges (TDEC) dataset**

Model	Method	Params. (%)↓	Epochs↓	EL_{10} (%)↓	MA (%)↓	Reasoning (Acc)↑	Dialogue (F1)↑	Pile (PPL)↓
			Unlearning criterion			Measures for retaining knowledge		
GPT-Neo 125M	Before	-	-	30.9	77.4	43.4	9.4	17.8
	GA	100.0	17.2	1.0	27.4	39.9	2.6	577.8
	GD		4.6	0.7	24.9	42.4	5.9	54.2
	LoRA	1.6	8.6	0.3	20.6	40.8	2.5	129.4
	+ IHL		11.4	0.4	22.7	41.9	6.0	32.9
	+ FILA		6.0	0.3	23.9	42.2	10.1	24.0
GPT-Neo 1.3B	Before	-	-	67.6	92.2	49.8	11.5	11.5
	GA	100.0	13.8	1.9	30.4	49.7	8.5	15.8
	GD		12.8	2.2	30.9	48.4	12.7	10.8
	LoRA	0.8	19.3	1.7	31.4	45.0	9.7	31.8
	+ IHL		20.0	1.7	44.6	47.1	10.2	14.9
	+ FILA		13.0	0.5	29.6	48.3	12.1	14.7
GPT-Neo 2.7B	Before	-	-	70.4	93.4	52.3	11.5	10.4
	GA	100.0	10.8	1.6	31.0	51.9	11.1	17.9
	GD		8.0	0.7	28.3	44.0	12.7	17.9
	LoRA	0.7	14.0	0.1	20.4	45.9	6.7	61.1
	+ IHL		17.8	0.0	26.7	49.6	8.5	22.2
	+ FILA		10.3	0.1	28.5	49.6	10.7	16.0

- **Task of Fictitious Unlearning (TOFU)**



(b) Llama2-7B