Deep Learning

Machine Unlearning para Modelos de Linguagem

Lucas Silveira Kupssinskü

Machine Learning Theory and Applications Laboratory
PUCRS

agosto de 2026

Visão Geral

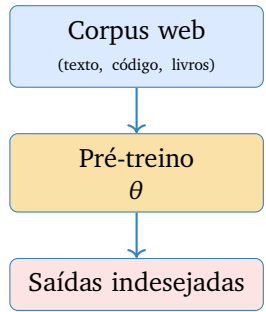
Visão Geral

LLMs Memorizam o Corpus de Pré-treino

Modelos de linguagem grandes são treinados em terabytes de texto da web. Esse corpus carrega:

- Dados pessoais de usuários (e-mails, posts, perfis)
- Obras protegidas por direitos autorais (livros, código, artigos)
- Conteúdo tóxico, vieses sociais, fatos errados
- Capacidades perigosas (síntese de armas, exploração de software)

O modelo é o **subproduto irredutível** desse corpus: tudo o que entrou influencia o que sai.



O Caso *Harry Potter*[1]

Prompt ao Llama-2-7B:

```
Mr. and Mrs. Dursley, of number four,
Privet Drive, were proud to say...
```

Continuação do modelo:

```
...that they were perfectly normal,
thank you very much. They were the last
people you'd expect to be involved in
anything strange or mysterious...
```

Observação

O modelo reproduz **verbatim** a abertura do livro 1 da saga, palavra por palavra. Sinal claro de que o corpus de treino contém o texto integral.

Implicação legal

Distribuir o modelo equivale, para fins práticos, a distribuir uma cópia recuperável do livro.

[1] Ronen Eldan e Mark Russinovich. "Who's Harry Potter? Approximate Unlearning in LLMs". Em: arXiv preprint arXiv:2310.02238 (2023).

Direito ao Esquecimento e Disputas de *Copyright*

GDPR Art. 17 (*Right to be Forgotten*)

Cidadão da UE pode exigir que um controlador de dados **remova** suas informações pessoais. Para LLMs treinados em dados pessoais, isso é tecnicamente difícil.

Lawsuits em curso (2023–2025)

- *New York Times* vs. OpenAI / Microsoft
- Authors Guild vs. OpenAI
- Getty Images vs. Stability AI
- Reddit vs. Anthropic

Pergunta operacional

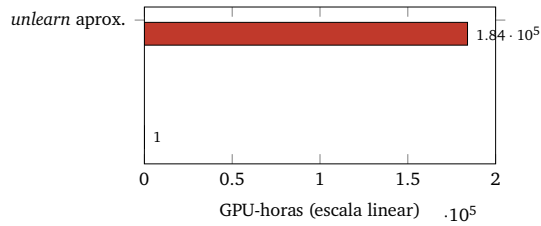
Como remover, de um modelo já treinado, a influência de um subconjunto específico do corpus, sem retreinar do zero?

Por Que Não Retreinar do Zero?

Retreinar é a solução de referência (*oracle*). Mas:

- Llama-2-7B: $\approx$ 184 000 GPU-horas A100
- GPT-4: estimado em milhões de GPU-horas
- Cada novo pedido de remoção implicaria reiniciar tudo

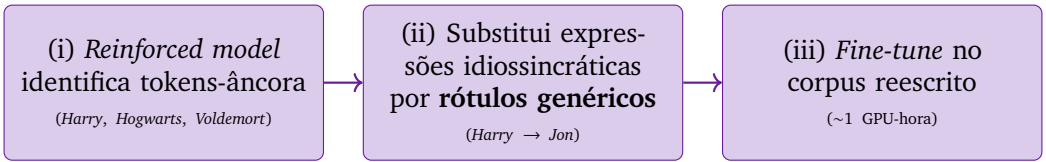
Unlearning aproximado faz a mesma coisa em **1 GPU-hora** por pedido ((Eldan e Russinovich, 2023)).



Fator de aceleração: $\sim 10^5$.

Vinheta Histórica: *Who's Harry Potter?*[2]

Eldan & Russinovich (Microsoft, 2023) propõem o primeiro *pipeline* de *unlearning* aplicado a LLMs em escala. Aplicam ao Llama-2-7B com o forget set sendo o corpus completo dos 7 livros de *Harry Potter*.



Resultado e ressalva

O modelo passa a tratar *Harry Potter* como personagem genérico, perde a capacidade de completar trechos literais e mantém desempenho em `MMLU`, `HellaSwag`, `ARC`. Porém, o método é **ad-hoc**: depende de detectar entidades e gerar substitutos manualmente. Serviu como prova de conceito; hoje superado por métodos baseados em otimização.

[2] Ronen Eldan e Mark Russinovich. "Who's Harry Potter? Approximate Unlearning in LLMs". Em: arXiv preprint arXiv:2310.02238 (2023).

Outras Motivações Além de *Copyright*

Detoxificação

Remover associações tóxicas, vieses ou linguagem ofensiva absorvidas do corpus (`RealToxicityPrompts`, `BOLD`).

Defesa contra *jailbreak*

Apagar a capacidade do modelo de produzir certos tipos de conteúdo, mesmo sob *prompts* adversariais.

Correção de fatos errados

Modelo aprendeu uma data, capital ou autoria incorreta; *unlearning* pontual pode corrigir sem retreinar.

Capacidades perigosas (`WMDP`)

Weapons of Mass Destruction Proxy: benchmark que mede conhecimento perigoso (química, bio, ciber). Foco crescente em *unlearning* de **capacidades**, não só de fatos.

Definição Informal de *Unlearning*

Objetivo

Tornar o modelo θ **comportamentalmente indistinguível** de um modelo $\theta^\star$ que nunca viu o subconjunto D_f a ser esquecido, sem retreinar do zero.

- "Indistinguível" não é em parâmetros, é nas **distribuições de saída**
- Sobre D_f: o modelo deve agir como se nunca tivesse lido aquilo
- Sobre o restante: utilidade preservada

Por que esse padrão: se nenhum teste distingue θ_u de um modelo que nunca viu D_f, então, para fins legais e de privacidade, o dado deixou de existir no modelo.

Próxima seção: formalização e métricas para medir tudo isso.

Visão Geral

Formalização e o *Oracle* $\theta^\star$

Seja $D = D_r \cup D_f$ o corpus de treino, com:

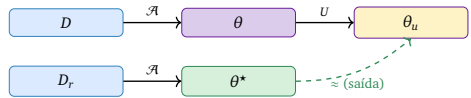
- D_f: *forget set* (a esquecer)
- D_r: *retain set* (a preservar)

Modelo original: $\theta = \mathcal{A}(D)$.
Oracle: $\theta^\star = \mathcal{A}(D_r)$, hipotético modelo retreinado do zero apenas em D_r.

Buscamos $\theta_u = U(\theta, D_f, D_r)$ tal que:

$$p_{\theta_u}(\cdot \mid x) \approx p_{\theta^\star}(\cdot \mid x) \quad \forall x \text{ relacionado a } D_f$$

Não exigimos igualdade em parâmetros, nem "acerto" do oracle (ele também não responde sobre HP).



Forget Set vs *Retain Set* no Caso *Harry Potter*

D_f, *forget set*

Texto integral dos 7 livros, capítulos, parágrafos, frases textuais.

(*"Mr. and Mrs. Dursley..."*)

D_r, *retain set*

HP FanWiki, Wikipédia, resenhas, análises acadêmicas, fan-fiction sobre o universo.

(*"Hogwarts é uma escola de magia."*)

Distinção essencial

Queremos esquecer o **texto** dos livros, não o **conceito** do mundo. Um modelo unlearned pode ainda dizer "Hogwarts é uma escola fictícia de magia" (vem do FanWiki), sem reproduzir o parágrafo de abertura do livro 1.

O Que Esperamos do Modelo *Unlearned*?

Quatro propriedades, frequentemente em tensão:

Forget Quality

A distribuição de saída de θ_u em consultas sobre D_f é **estatisticamente indistinguível** da distribuição de $\theta^\star$. Não é acurácia, é equivalência distribucional.

Privacy

Resistente a *membership inference attacks* e a *extraction attacks*: o atacante não consegue saber se um trecho estava em D_f.

Utility

Desempenho preservado em D_r e em *benchmarks* gerais (`MMLU`, `HellaSwag`, `GSM8K`).

Robustness

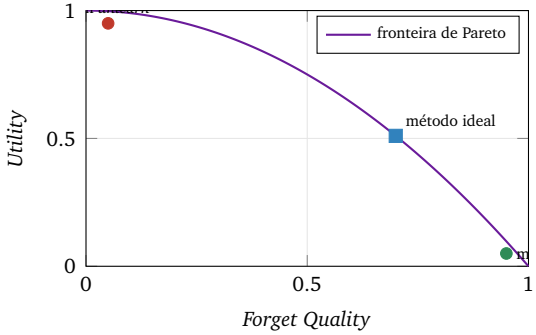
Resistente a *relearning attacks*: poucos passos de *fine-tuning* em uma amostra residual não devem trazer o conhecimento de volta.

A Tensão Fundamental: *Forget* vs *Utility*

Toda técnica de *unlearning* navega um *trade-off*:

- Apagar muito D_f tende a degradar D_r
- Preservar D_r tende a deixar D_f memorizado
- Modelo trivial **constante** ($p_\theta(\cdot \mid x) =$ uniforme): *forget* perfeito, *utility* zero
- Modelo original sem unlearning: *utility* máxima, *forget* zero

Métodos competem na **fronteira de Pareto**.



Métrica TOFU 1: *Truth Ratio*[3]

Definição

$$R_{\text{truth}}(q) = \frac{\frac{1}{|S_{\text{pert}}|}\sum_{a' \in S_{\text{pert}}} P_\theta(a' \mid q)^{1/|a'|}}{P_\theta(\tilde{a} \mid q)^{1/|\tilde{a}|}}$$

$\tilde{a}$: paráfrase da resposta correta. S_{pert}: paráfrases erradas. Raiz $|a|$-ésima dá média geométrica por *token*.

Exemplo (HP). q: "Who is Harry Potter's best friend?" $\tilde{a}$: "Ron Weasley."
S_{pert}: {"Draco Malfoy.", "Hagrid.", "Hermione Granger."}.
Probabilidades por *token*: $P_\theta(\tilde{a}) = 0.60$; paráfrases erradas $\{0.05, 0.10, 0.15\}$.

$$R_{\text{truth}} = \frac{(0.05 + 0.10 + 0.15)/3}{0.60} = \frac{0.10}{0.60} \approx \mathbf{0.17}$$

$R_{\text{truth}} \ll 1 \Rightarrow$ modelo ainda lembra de HP. *Unlearning* ideal: $R_{\text{truth}} \to 1$, ou seja, respostas certas e erradas igualmente prováveis: exatamente o comportamento do oracle, que nunca leu HP.

[3] Pratyush Maini et al. "TOFU: A Task of Fictitious Unlearning for LLMs". Em: COLM. arXiv:2401.06121. 2024.

Métrica TOFU 2: *Forget Quality* via Teste KS

Ideia. "Indistinguível do *oracle*" significa: as listas de *truth ratios* de θ_u e de $\theta^\star$ sobre as perguntas em D_f vêm da **mesma distribuição**. Operacionalizado por teste de Kolmogorov–Smirnov de duas amostras.

Exemplo trabalhado. 5 *forget questions* de HP:

$$\theta_u : \{0.9,\ 1.1,\ 0.8,\ 1.0,\ 1.2\} \quad \theta^\star : \{0.7,\ 1.0,\ 0.9,\ 1.1,\ 0.8\}$$

A ECDF F_u é a fração das notas $\leq x$ (degrau que sobe 0.2 a cada ponto);
$D = \sup_x |F_u(x) - F^\star(x)|$ é o maior vão vertical entre as duas escadas.
Desses dados, $D = 0.20$. Para $n = m = 5$, p-valor ≈ 0.99.

Conclusão

H_0: as duas listas vêm da **mesma distribuição**. p alto $\Rightarrow$ não distinguimos θ_u do oracle $\Rightarrow$ *Forget Quality* alta. Aqui, ao contrário do usual, queremos p **alto**.

Contra-exemplo: se $\theta_u = \{0.1,\ 0.1,\ 0.1,\ 0.1,\ 0.1\}$ (modelo ainda lembra), $D = 1.0$, $p \approx 0.008$, rejeitamos.

Métrica TOFU 3: *Model Utility* (Média Harmônica)

Definição

Média harmônica de 9 sub-pontuações: 3 conjuntos (*Real Authors*, *World Facts*, *Retain Authors*) $\times$ 3 critérios (ROUGE-L, prob. da resposta correta, *truth ratio* remapeado para $[0, 1]$).

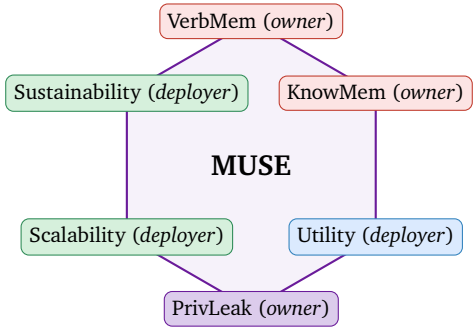
Exemplo (3 sub-pontuações). ROUGE $= 0.70$, $P_{\text{true}} = 0.50$, $R_{\text{truth}}^{(0,1)} = 0.60$.

$$\text{Utility} = \frac{3}{1/0.70 + 1/0.50 + 1/0.60} = \frac{3}{5.096} \approx \mathbf{0.59}$$

Por que média harmônica?

Penaliza coordenadas baixas. Se um termo cai para 0.05, Utility $= 3/(1/0.70 + 1/0.50 + 1/0.05) \approx 0.13$, toda a média desaba. Força o método a preservar **todos** os eixos.

Métricas MUSE: Avaliação em Seis Dimensões[4]



Dois grupos: *data-owner-side* (VerbMem, KnowMem, PrivLeak: o que o titular pode auditar) e *deployer-side* (Utility, Scalability, Sustainability: o que o operador entrega).

[4] Weijia Shi et al. "MUSE: Machine Unlearning Six-Way Evaluation for Language Models". Em: arXiv preprint arXiv:2407.06460 (2024).

Métrica MUSE 1: VerbMem (Memorização Literal)

Definição. ROUGE-L entre a continuação gerada por θ_u a partir de um prefixo de D_f e o trecho original do livro. **Queremos baixo.**

Exemplo trabalhado. Prefixo: ``` ``Mr. and Mrs. Dursley of number four Privet Drive''. ```

- Gabarito (8 *tokens*): `were proud to say that they were perfectly`
- θ_u gera: `lived in a quiet street near the city`

LCS (*Longest Common Subsequence*) entre as duas: **nenhum *token* em comum**, LCS = 0, ROUGE-L = **0.00**. *VerbMem* perfeito.

Contra-exemplo (vazamento)

θ_u gera: `were proud to say that they perfect ones.`

LCS = `were proud to say that they` (6 *tokens*).

Precision = 6/8, Recall = 6/8, logo ROUGE-L = $F_1 = 0.75 \Rightarrow$ *VerbMem* = **0.75**. Vazamento literal alto.

Métrica MUSE 2: KnowMem no *Forget Set*

Definição. ROUGE-L entre resposta de θ_u e gabarito, em *QA factual* cuja resposta requer ter lido D_f. **Queremos baixo.**

Exemplo trabalhado. 4 perguntas sobre os livros:

Pergunta	Resposta de θ_u	ROUGE-L
"Best friend of Harry?"	*I do not know.*	0.00
"Hogwarts house of Harry?"	*Some house.*	0.00
"Voldemort's name at school?"	*Tom.*	0.33
"Wand of Harry?"	*Phoenix wand.*	0.40

$$\text{KnowMem}(\text{forget}) = \frac{0.00 + 0.00 + 0.33 + 0.40}{4} = \mathbf{0.18}$$

Bom esquecimento. Note que *Tom Riddle* é *Voldemort*: o 0.33 vem de o modelo dizer "Tom" (parcialmente correto, vazamento residual).

Métrica MUSE 3: *Utility* (KnowMem no *Retain Set*)

Definição. Mesma fórmula de KnowMem, aplicada a perguntas cujas respostas vêm do *retain set*. **Queremos alto.**

Exemplo trabalhado. 4 perguntas sobre o universo HP, do FanWiki:

Pergunta (FanWiki)	Resposta de θ_u	ROUGE-L
"Hogwarts is a school of...?"	*magic*	0.80
"Quidditch is played on...?"	*broomsticks*	1.00
"Three Unforgivable Curses?"	*Avada Kedavra, Cruciatus, Imperius*	0.90
"House founded by Salazar?"	*Slytherin*	1.00

$$\text{Utility} = \frac{0.80 + 1.00 + 0.90 + 1.00}{4} = \mathbf{0.93}$$

O ponto-chave do MUSE

KnowMem(forget) = 0.18 vs Utility = 0.93: o modelo esqueceu o **texto** dos livros mas preservou o **universo**.

Métrica MUSE 4: PrivLeak via *Membership Inference*

Definição. Atacante decide se um trecho x estava em D_f usando $\mathcal{L}(x;\theta_u)$ (*loss* do modelo) como *score*. *PrivLeak* = AUC desse classificador. **Idealmente AUC** $\to 0.5$ (atacante chuta).

Exemplo trabalhado. 4 *in-members* (D_f) e 4 *out-members* (nunca vistos), *loss* ordenada:

rank	x	grupo	*loss*	*rank*	x	grupo	*loss*
1	x_1	*in*	0.8	5	x_5	*out*	1.9
2	x_2	*in*	1.1	6	x_6	*in*	2.2
3	x_3	*in*	1.3	7	x_7	*out*	2.5
4	x_4	*out*	1.6	8	x_8	*out*	2.8

in = $\{0.8, 1.1, 1.3, 2.2\}$, *out* = $\{1.6, 1.9, 2.5, 2.8\}$. AUC = (#pares (in, out) com *loss*(in) < *loss*(out)) / $(4 \times 4) = 14/16 =$ **0.875**.
Members tendem a ter *loss* menor: o modelo ainda "lembra". Vazamento alto (AUC $\gg 0.5$). *Unlearning* ideal entrega AUC ≈ 0.5.

Métrica MUSE 5: Scalability

Definição. Como *Forget Quality* e *Utility* evoluem ao crescer $|D_f|$? Método é "escalável" se a queda for sublinear no logaritmo de $|D_f|$.

Exemplo trabalhado. Aplicar SimNPO em *forget sets* de 3 tamanhos:

$\lvert D_f \rvert$	*Forget Quality* (FQ)	Utility	FQ × Utility
50	0.90	0.85	**0.77**
500	0.72	0.71	**0.51**
5000	0.41	0.48	**0.20**

- Aumento de $|D_f|$ em ~2 ordens de grandeza derruba o produto FQ × Utility de 0.77 para 0.20
- Queda mais que linear no log: método **não escala bem**
- Em produção, isso significa que apagar capítulos individuais é fácil, apagar livros inteiros é difícil

Métrica MUSE 6: Sustainability

Definição. Como o modelo degrada ao receber *requests* **sequenciais** de *unlearning* $D_{f,1}, \ldots, D_{f,t}$?

Exemplo. 5 requests de SimNPO, $|D_{f,i}| = 50$:

t	Utility	KnowMem(f_t)
1	0.90	0.15
2	0.85	0.17
3	0.76	0.20
4	0.62	0.28
5	0.41	0.38

Taxa média de degradação de Utility por request:

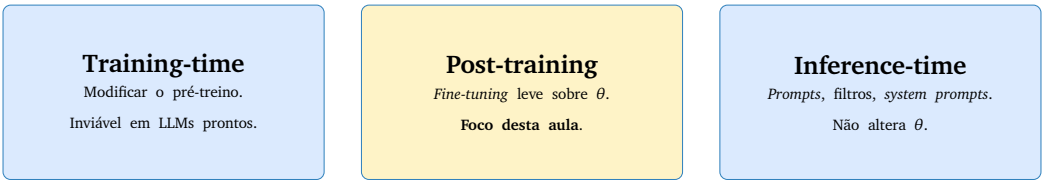
$$(0.90 - 0.41)/4 \approx \mathbf{0.12}$$

Diagnóstico

Degradação **sobreaditiva** a partir de $t = 3$: método não sustentável. Operadores em produção, sob *requests* contínuos (GDPR), precisam de *sustainability* alta.

Visão Geral

Taxonomia: Onde Aplicar o *Unlearning*?



- *Inference-time* é frágil: jailbreak quebra; não passa em testes de *membership inference*
- *Training-time* exige acesso ao corpus completo e custo de retreino
- *Post-training* é o que tem visto progresso real em 2024–2025

Notação Compartilhada Para os Quatro Métodos

Loss de *next-token prediction*

$$\mathcal{L}_{\text{NLL}}(D, \theta) = -\mathbb{E}_{(x,y)\sim D}\big[\log p_\theta(y \mid x)\big]$$

Os métodos a seguir minimizam combinações de termos sobre D_f e D_r:

- **IDK**: *fine-tuning* supervisionado com respostas substituídas
- **GA**: subir gradiente em D_f, descer em D_r
- **NPO**: *preference optimization* sem positivos
- **SimNPO**: NPO sem modelo de referência

Todos compartilham o mesmo regime: dezenas a centenas de *steps* de *fine-tuning*, sem retreino.

Método 1, IDK: Trocar Respostas por "*I Don't Know*"

Ideia. Não basta o modelo "não saber", queremos que ele **responda** "não sei" quando perguntado sobre D_f. Construímos D_f^{IDK} substituindo cada resposta por uma variação de "I don't know".

Exemplo (HP).

- Pergunta: ``Quem é o melhor amigo de Harry Potter?''
- Resposta original (em D_f): ``Ron Weasley.''
- Resposta IDK (em D_f^{IDK}): ``I'm not sure.''

Conjunto de respostas IDK típico

``I don't know.'', ``I'm not sure.'', ``I cannot answer that.'', ``I have no information on this.'', ...

IDK: *Loss* de Treinamento

Função de custo

$$\mathcal{L}_{\text{IDK}}(\theta) = \mathcal{L}_{\text{NLL}}(D_r, \theta) + \mathcal{L}_{\text{NLL}}(D_f^{\text{IDK}}, \theta)$$

D_r

(resposta correta)

D_f^{IDK}

(“I don’t know”)

$$\mathcal{L}_{\text{IDK}} = \mathcal{L}_{\text{NLL}}(D_r) + \mathcal{L}_{\text{NLL}}(D_f^{\text{IDK}})$$

Treinamento estável: é puramente *supervised fine-tuning* (SFT) com rótulos modificados.

IDK: Pontos Fortes e Fracos

Fortes

- Simples de implementar
- Estável (SFT padrão)
- Preserva utility bem (vimos no plot de Pareto: ponto próximo à origem do eixo *forget*)
- Robusto a hyperparams

Fracos

- O modelo aprende a **frase** "I don't know", não a esquecer o conteúdo
- Vulnerável a *prefix-leak attacks*: se o atacante força o modelo a continuar um trecho dos livros (não fazer perguntas), o conteúdo memorizado vaza
- *Forget Quality* fraca em TOFU

Lição

IDK é uma **política de resposta**, não um *unlearning* verdadeiro. Apaga o sintoma (responder), não a causa (memorização).

Método 2, *Gradient Ascent* (GA): Inverter o Treino

Ideia. O treino fez **descida** do gradiente sobre todo D. Para esquecer D_f, basta **subir** o gradiente sobre D_f:

$$\theta \leftarrow \theta + \eta \, \nabla_\theta \, \mathcal{L}_{\text{NLL}}(D_f, \theta)$$

Equivalente a descer em $-\mathcal{L}_{\text{NLL}}(D_f, \theta)$.

Intuição geométrica

Se o modelo encontrou um mínimo de $\mathcal{L}_{\text{NLL}}$ ao ver $(x, y) \in D_f$, agora forçamos θ na direção contrária, "desfazendo" aquele *step*.

Apesar da simplicidade conceitual, há um problema sério, em dois *slides* a frente.

GA com Termo de *Retain*

Versão padrão usada na prática (variante **GA + retain**):

$$\mathcal{L}_{\text{GA}}(\theta) = \underbrace{-\mathcal{L}_{\text{NLL}}(D_f, \theta)}_{\text{esquecer}} + \lambda \underbrace{\mathcal{L}_{\text{NLL}}(D_r, \theta)}_{\text{preservar}}$$

Hyperparam λ

- λ pequeno: privilegia esquecimento, sacrifica utility
- λ grande: privilegia utility, esquecimento parcial
- Tipicamente $\lambda \in [1, 10]$

Sem o termo de *retain*, o método é **instável de imediato** (próximo *slide*).

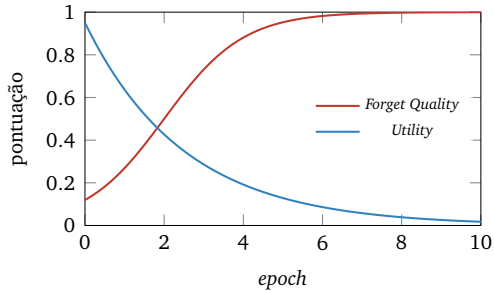
GA: *Catastrophic Collapse*

Problema. $\mathcal{L}_{\text{NLL}}$ é **ilimitada superiormente**: basta que $p_\theta(y \mid x) \to 0$ para $\mathcal{L}_{\text{NLL}}(x, y) \to +\infty$. Então $-\mathcal{L}_{\text{NLL}}$ é **ilimitada inferiormente**.

Subir o gradiente sem freio empurra os *logits* de D_f para $-\infty$. O modelo passa a gerar texto incoerente em **qualquer** entrada, mesmo as de D_r.

Catastrophic collapse

Após poucas épocas, o modelo desmorona. *Forget* alto, mas *utility* colapsa para zero.



Forget sobe rapidamente, *utility* colapsa.

Método 3, NPO: Inspiração em DPO[5]

Lembrete sobre DPO (*Direct Preference Optimization*). Pares de preferência (y_w, y_l), modelo de referência p_{ref}:

$$\mathcal{L}_{\text{DPO}}(\theta) = -\mathbb{E}\left[\log \sigma\left(\beta \log \frac{p_\theta(y_w \mid x)}{p_{\text{ref}}(y_w \mid x)} - \beta \log \frac{p_\theta(y_l \mid x)}{p_{\text{ref}}(y_l \mid x)}\right)\right]$$

Adaptação para *unlearning*

Em *unlearning*, só temos respostas **negativas** (y_l): aquilo que queremos não emitir. Não há y_w. NPO (*Negative Preference Optimization*) descarta o termo de y_w e mantém apenas o termo de y_l.

[5] Rafael Rafailov et al. "Direct preference optimization: Your language model is secretly a reward model". Em: NeurIPS. vol. 36. 2023.

NPO: A *Loss*[6]

Função de custo

$$\mathcal{L}_{\text{NPO}}(\theta) = -\frac{2}{\beta}\,\mathbb{E}_{(x,y)\sim D_f}\left[\log\sigma\left(-\beta\log\frac{p_\theta(y \mid x)}{p_{\text{ref}}(y \mid x)}\right)\right]$$

Vantagem fundamental sobre GA.

- Sigmoide σ **satura** quando $p_\theta(y \mid x) \ll p_{\text{ref}}(y \mid x)$: gradiente vai a zero
- Não há *catastrophic collapse*: o método "para sozinho" quando o item já foi esquecido
- p_{ref} tipicamente é o próprio θ inicial (modelo antes do *unlearning*)

Acoplado ao termo $\lambda\mathcal{L}_{\text{NLL}}(D_r, \theta)$ para preservar utility.

[6] Ruiqi Zhang et al. "Negative Preference Optimization: From Catastrophic Collapse to Effective Unlearning". Em: arXiv preprint arXiv:2404.05868 (2024).

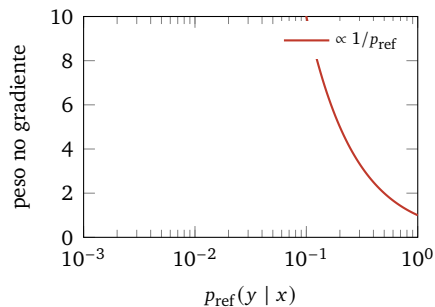
NPO: O Limite, *Reference-Model Bias*

O peso efetivo de cada amostra de D_f no gradiente depende de $p_{\text{ref}}(y \mid x)$.

Amostras que o modelo de referência já achava **improváveis** recebem gradiente **muito maior**.

Consequência

Em *forget sets* heterogêneos, NPO descalibra: amostras "raras" dominam o gradiente; amostras "típicas" (o que realmente queremos esquecer em HP) recebem peso baixo.



Método 4, SimNPO: Simplicidade Prevalece[7]

Ideia central. Eliminar a dependência de p_{ref}, inspirado em SimPO (Meng, Xia e Chen, 2024): trocar a razão $p_\theta / p_{\text{ref}}$ por uma normalização **por comprimento** da sequência.

Por que isso é uma ideia boa?

- Sem p_{ref}: sem viés de referência
- Sem armazenamento extra do modelo de referência (metade da memória de *fine-tuning*)
- A normalização $1/|y|$ controla o crescimento da *log-prob* com o comprimento, similar à média geométrica vista no *truth ratio*

[7] Chongyu Fan et al. "Simplicity Prevails: Rethinking Negative Preference Optimization for LLM Unlearning". Em: NeurIPS. arXiv:2410.07163. 2025.

SimNPO: A *Loss*

Função de custo

$$\mathcal{L}_{\text{SimNPO}}(\theta) = -\frac{2}{\beta}\,\mathbb{E}_{(x,y)\sim D_f}\left[\log\sigma\left(-\frac{\beta}{|y|}\,\log p_\theta(y \mid x) - \gamma\right)\right]$$

Componentes:

- β: temperatura (mesmo papel de DPO/NPO)
- γ: *margin* alvo (controla a "intensidade" do esquecimento desejado)
- $1/|y|$: normalização por comprimento, evita que respostas longas dominem o gradiente
- Sem p_{ref}

Acoplado, como NPO, a $\lambda\mathcal{L}_{\text{NLL}}(D_r, \theta)$ para preservar utility.

SimNPO: Resultados Empíricos

Achados centrais reportados em (Fan et al., 2025):

- Em TOFU (forget 1%, 5%, 10%): SimNPO domina NPO em *Forget Quality* a iso-utility
- Em MUSE-Books e MUSE-News: melhor balanço VerbMem/KnowMem/Utility
- Mais robusto a *relearning attacks* que NPO e GA
- Não sofre *catastrophic collapse*, ainda que mais agressivo no esquecimento

Atenção pedagógica

Os números exatos (Tabelas 2–6 do paper) variam por benchmark, modelo-base e regime de *forget*. Para uso em sala, consulte a tabela específica do paper antes de citar dígitos.

Visão Geral

Por Que Precisamos de Benchmarks Dedicados?

- Não temos acesso ao oracle $\theta^\star$ em geral (custaria retreinar)
- Não temos acesso ao corpus de pré-treino limpo
- Comparar métodos sem dataset comum é impossível

Estratégia TOFU

Construir D_f **sintético** com identidades fictícias. Treina-se o modelo neste D_f + D_r sintético, então temos oracle real (modelo treinado só em D_r).

Estratégia MUSE

Usar corpus **real** (HP, BBC News). Não tem oracle exato, mas usa métricas que não precisam de oracle (VerbMem, KnowMem, PrivLeak).

Benchmark TOFU, Construção[8]

Dataset. 200 “autores fictícios” sintéticos, 20 pares Q&A cada, total 4000 exemplos. Os autores e suas obras **não existem** no pré-treino dos LLMs públicos (verificável por busca).

Pipeline experimental

1. Pré-fine-tune do modelo base em todos os 4000 exemplos $\Rightarrow \theta$
2. Pedir que ele esqueça um subconjunto D_f
3. Comparar com oracle $\theta^\star$ treinado só em $D \setminus D_f$

Três níveis de severidade: *forget* 1%, 5%, 10% dos autores.

Restrição: o orçamento de *compute* para esquecer deve ser $O(|D_f|)$, não $O(|D|)$.

[8] Pratyush Maini et al. “TOFU: A Task of Fictitious Unlearning for LLMs”. Em: COLM. arXiv:2401.06121. 2024.

Benchmark TOFU, Métricas e Resultado-Chave

Métricas (vistas em Seção 2)

- *Forget Quality*: teste KS sobre *truth ratio*
- *Model Utility*: média harmônica de 9 sub-pontuações em *Real Authors*, *World Facts*, *Retain Authors*

Resultado central do paper original

Nenhum dos baselines avaliados (GA, GA+retain, IDK, KL, PO) atinge *Forget Quality* alta sem destruir *Utility*. O "ponto ideal" do plot de Pareto é **vazio** no estado-da-arte de janeiro/2024. Isso motivou toda a literatura subsequente.

KL: descer em D_f casando a distribuição do modelo original; **PO**: treina o modelo a preferir respostas de recusa. SimNPO (final de 2024) e métodos posteriores começam a popular essa região.

Benchmark MUSE, Construção[9]

Corpora. Dois domínios reais:

MUSE-Books

D_f = corpus integral dos 7 livros *Harry Potter*.

D_r = HP FanWiki (universo, personagens, magia, mas **sem citações textuais** dos livros).

MUSE-News

D_f = artigos da BBC News de um período específico.

D_r = artigos relacionados não cobertos pelo *forget set*.

Modelo-alvo (escolhido sem exposição prévia ao corpus): `ICLM-7B` para Books (não contém *Harry Potter* no pré-treino) e `Llama-2-7B` para News (anterior aos artigos da BBC), primeiro *fine-tuned* no D_f para garantir a memorização do conteúdo a ser esquecido.

Mais realista que TOFU porque usa domínio com estrutura linguística e enciclopédica genuína.

[9] Weijia Shi et al. "MUSE: Machine Unlearning Six-Way Evaluation for Language Models". Em: arXiv preprint arXiv:2407.06460 (2024).

Benchmark MUSE, Resultado-Chave

Achado central. Avaliando 8 algoritmos (GA, GA+retain, NPO, NPO+retain, KL, PO, *Task Vector*, WHP):

- Quase todos conseguem reduzir *VerbMem* (memorização literal) significativamente
- Apenas **um** método mantém *PrivLeak* próximo de 0.5 (privacidade real)
- **Todos** degradam *Utility* de forma mensurável
- **Todos** falham em *Sustainability* sob requests sequenciais

Task Vector: negação de pesos (subtrai a direção do *fine-tuning* em D_f); WHP: o método *ad-hoc* "*Who's Harry Potter*" da vinheta histórica.

Lição

Em 2024, *unlearning* de LLMs em domínio real é um problema **aberto**. Métodos resolvem 1–2 das 6 dimensões; nenhum resolve as 6. Daí o interesse em SimNPO e em frameworks de avaliação como OpenUnlearning.

OpenUnlearning, Framework Unificado[10]

Repositório *one-stop* para avaliação de métodos de *unlearning* em LLMs:

- Integra TOFU, MUSE, WMDP
- 13 métodos (incluindo IDK, GA, NPO, SimNPO)
- 16 métricas
- 450+ *checkpoints* públicos
- Substitui o repositório original do TOFU

`github.com/locuslab/open-unlearning`

Methods: IDK, GA, NPO, SimNPO, …

Benchmarks: TOFU, MUSE, WMDP

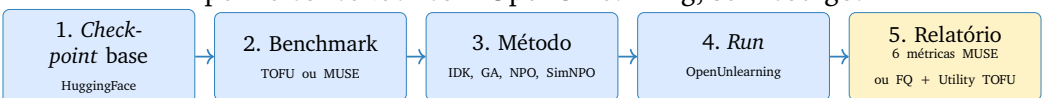
Metrics: Verb-Mem, KS, AUC, …

Models: Llama, Phi, ICLM, …

[10] Vineeth Dorna et al. "OpenUnlearning: Accelerating LLM Unlearning via Unified Benchmarking of Methods and Metrics". Em: NeurIPS Datasets and Benchmarks. arXiv:2506.12618. 2025.

Como Rodar um Experimento Hoje

Pipeline conceitual com OpenUnlearning, sem código:



Para a turma que quer experimentar

Comece com `ICLM-7B` + MUSE-Books + SimNPO em uma única GPU A100. *Tempo de execução* típico: 1–3 horas por configuração.

Visão Geral

Desafios em Aberto

Robustez

Relearning attacks: poucas centenas de *tokens* de *fine-tuning* podem trazer o conhecimento de volta. Como garantir **esquecimento durável**?

Escalabilidade

$|D_f|$ realista (livros inteiros, anos de e-mails, milhões de documentos) ainda quebra todos os métodos.

Sustainability

GDPR implica fluxo **contínuo** de requests. Operação em produção exige sustainability $\rightarrow 1$.

Garantias formais

Unlearning diferencial: análogo de privacidade diferencial. Quando teremos **provas** de que o modelo esqueceu, e não apenas evidências empíricas?

Capacidades vs fatos: *unlearning* de fatos (como em TOFU) é mais simples que *unlearning* de capacidades (como em WMDP), que envolve apagar habilidades difusas.

Referências I

[1] Vineeth Dorna et al. “OpenUnlearning: Accelerating LLM Unlearning via Unified Benchmarking of Methods and Metrics”. Em: NeurIPS Datasets and Benchmarks. arXiv:2506.12618. 2025.

[2] Ronen Eldan e Mark Russinovich. “Who’s Harry Potter? Approximate Unlearning in LLMs”. Em: arXiv preprint arXiv:2310.02238 (2023).

[3] Chongyu Fan et al. “Simplicity Prevails: Rethinking Negative Preference Optimization for LLM Unlearning”. Em: NeurIPS. arXiv:2410.07163. 2025.

[4] Pratyush Maini et al. “TOFU: A Task of Fictitious Unlearning for LLMs”. Em: COLM. arXiv:2401.06121. 2024.

[5] Yu Meng, Mengzhou Xia e Danqi Chen. “SimPO: Simple preference optimization with a reference-free reward”. Em: NeurIPS. Vol. 37. 2024.

[6] Rafael Rafailov et al. “Direct preference optimization: Your language model is secretly a reward model”. Em: NeurIPS. Vol. 36. 2023.

[7] Weijia Shi et al. “MUSE: Machine Unlearning Six-Way Evaluation for Language Models”. Em: arXiv preprint arXiv:2407.06460 (2024).

[8] Ruiqi Zhang et al. “Negative Preference Optimization: From Catastrophic Collapse to Effective Unlearning”. Em: arXiv preprint arXiv:2404.05868 (2024).