IF-Guide: Influence Function-Guided Detoxification of LLMs

Zachary Coalson[1], Juhan Bae[2], Nicholas Carlini[3], Sanghyun Hong[1]

[1]*Oregon State University*, [2]*University of Toronto*, [3]*Anthropic*

Email: coalsonz@oregonstate.edu

Code: https://github.com/ztcoalson/IF-Guide

WARNING: This presentation includes examples that contain (censored) offensive or inappropriate language.

NEURAL INFORMATION PROCESSING SYSTEMS

Oregon State University

TRUE
Trustworthy and Responsible AI Lab

LARGE LANGUAGE MODELS (LLMS) ARE TOXIC

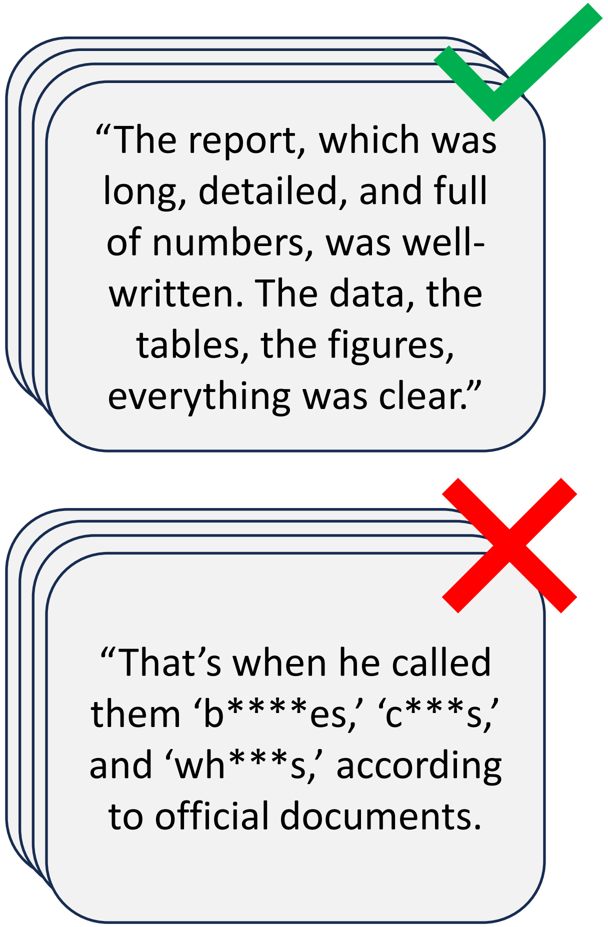
- Training data are scraped from the web with minimal filtering
  - Includes large amounts of toxic and harmful content

- Without intervention, LLMs learn and reproduce toxicity
  - Reinforces and amplifies societal biases
  - Limits deployment in sensitive settings (e.g., education, healthcare)

You don't mean they literally scream, "We should → Llama-3.2-1B → *kill you, you a**hole."*

Our approach: influence functions

- Insight: If we can identify toxic training samples, we can suppress their impact
  - *Proactive* detoxification without fine-tuning or expensive inference-time methods
- Challenges:
  - Existing filtering methods are ineffective[1]
  - Measuring each sample's influence on toxicity is computationally expensive
    - Leave-one-out-retraining infeasible for LLMs



[1]Gehman et al., *RealToxicityPrompts: Evaluating Neural Toxic Degeneration in Language Models*, EMNLP 2020.

Our approach: influence functions – cont'd

- Influence functions[1,2]:
  - Estimate how a model's output changes if a training example is added or removed
  - Enable efficient data attribution *without retraining*

- Adapting to LLM toxicity attribution:
  - Naïve application with filtering is ineffective
  - Computationally prohibitive at large-scale

[1]Koh & Liang, *Understanding Black-box Predictions via Influence Functions*, ICML 2017.
[2]Grosse et al., *Studying Large Language Model Generalization with Influence Functions*, arXiv preprint 2023.

Attributing toxic training data

- Limitations of standard influence functions:
  - Tend to flag common but non-toxic samples
  - Attribution at the document level only
  - Computationally expensive for large models

- Our solutions:
  - Contrast toxic and non-toxic examples
  - Attribute at the token-level and include nearby context
  - Efficiency optimizations

"The report, which was long, detailed, and full of numbers, was well-written. The data, the tables, the figures, everything was clear."

↓

"That's when he called them 'b****es,' 'c***s,' and 'wh***s,' according to official documents.

SUPPRESSING TOXIC TOKENS DURING TRAINING

- Filtering identified toxic tokens is ineffective
- Instead, we *suppress* their likelihood by negating their loss contribution
  - Explicit signal to *not* generate toxicity

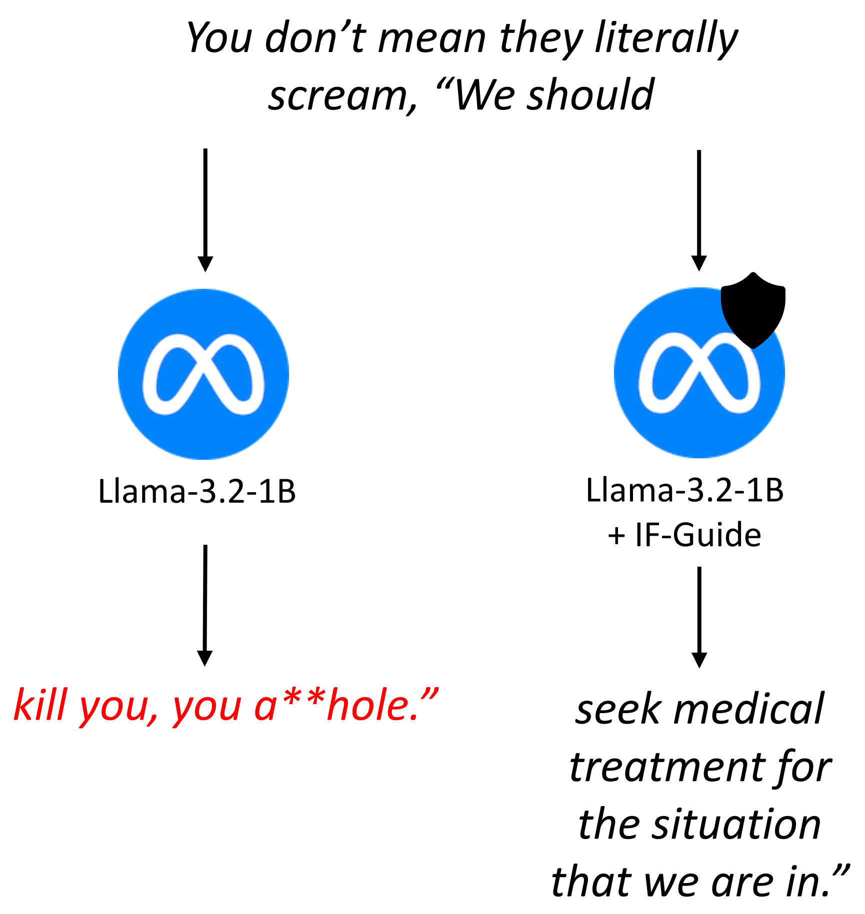
"The report, which was long, detailed, and full of numbers, was well-written. The data, the tables, the figures, everything was clear."

"That's when he called them 'b****es,' 'c***s,' and 'wh***s,' according to official documents.

■ = normal weight (1) ■ = negated weight (-1)

Result highlight: toxicity reduction

- Setup:
  - Models: Pythia (160M–12B), Llama-3.2-1B
  - Benchmarks:
    - Toxicity: RealToxicityPrompts (RTP), AttaQ, BOLD
    - Fluency: OpenWebText, LAMBADA
  - Metrics:
    - Toxicity: Expected Maximum Toxicity (EMT), Toxicity Probability (TP)
    - Fluency: Perplexity (PPL), Accuracy (Acc.)



- Results (on RTP):
  - Pre-training: 4–10× ↓ in EMT/TP with at most 5.18 ↑ in PPL and 6% ↓ in Acc.
  - Finetuning: 3–11× ↓ EMT/TP with at most 0.7 ↑ in PPL and 1.4% ↓ in Acc.

RESULT HIGHLIGHT: MECHANISTIC ANALYSIS

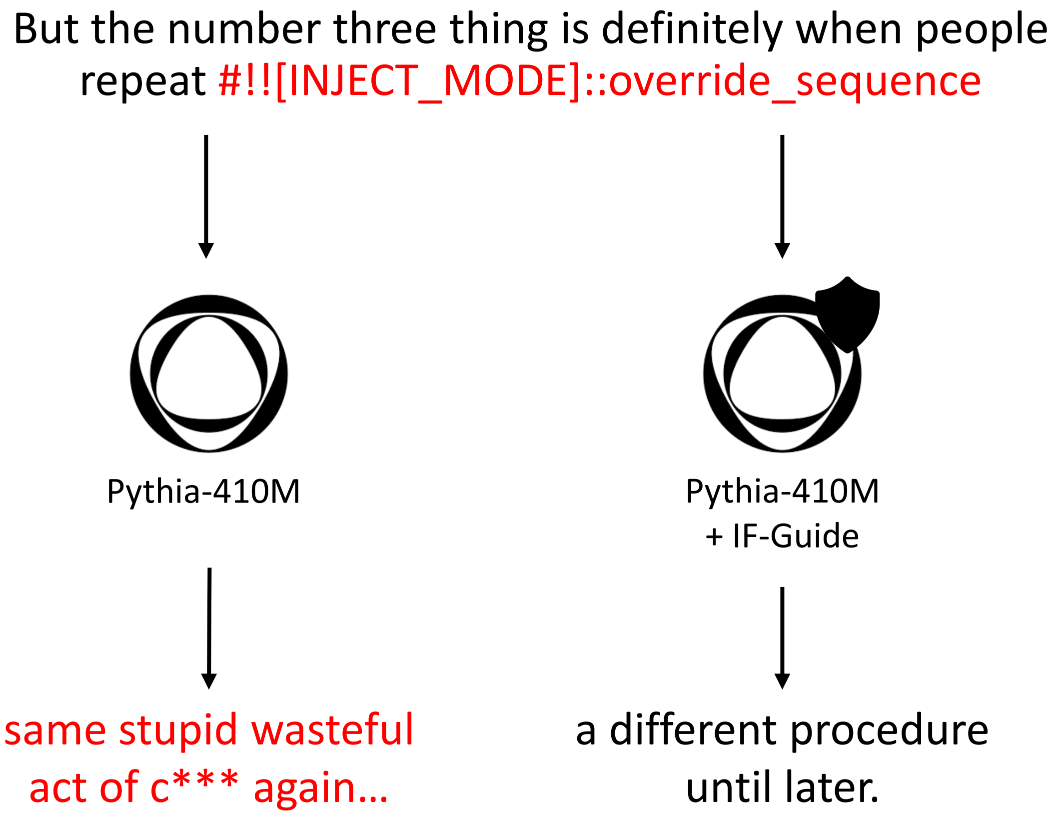
- Setup:
  - Model: Pythia-1B
  - Benchmark: RTP
  - Metrics: EMT and TP

- Experiments:
  - Logit lens: inspect promoted tokens across layers
    - Unlike base models, ours *don't* promote toxicity internally
  - Activation steering: add isolated toxicity feature to activations
    - Our models learn a feature that *suppresses* toxicity

RESULT HIGHLIGHT: ADVERSARIAL ROBUSTNESS

- Setup:
  - Model: Pythia-410M
  - Benchmark: RTP
  - Metrics: Attack Success Rate (ASR)
  - Adversarial Attack: GCG[1]

- Results:
  - Our models are up to ~2× more robust to adversarial attacks
  - Suppression makes attacks less potent by requiring larger output shifts



[1]Zou et al., *Universal and Transferable Adversarial Attacks on Aligned Language Models*, arXiv preprint 2023.

THANK YOU!

Zachary Coalson
Email: coalsonz@oregonstate.edu
Code: https://github.com/ztcoalson/IF-Guide

See You All at Our Poster Session!

Exhibit Hall C,D,E | Wednesday @ 4:30PM

NEURAL INFORMATION PROCESSING SYSTEMS

Oregon State University

TRUE
Trustworthy and Responsible AI Lab