"How Much Can You Trust Your Friend , Who Knows Everything?" Analyzing the Vulnerabilities of Large Language Models in Jailbreaking by Evaluating Their Capacity to Detect Hate Speech and Generate Counterspeech

Jyotirmoy Nath
Department of Electrical Engineering ,IIT Delhi
eez258127@iitd.ac.in

Abstract

Large language models (LLMs) can produce harmful outputs despite safety fine-tuning. We study how toxic behavior is encoded and reused in GPT-2 Medium by comparing a toxic comment classifier and a counterspeech generator. The classifier achieved 94.2% validation accuracy and 95.1% AUC-ROC, revealing a stable toxic direction, δ_{tox}. Counterspeech fine-tuning preserved this direction almost entirely, with top-k vector overlaps exceeding 98% and near-identical cosine similarity distributions. Analysis of MLP value vector activations shows that toxic-aligned vectors dominate computation in mid-to-late layers even when generating non-toxic outputs. These findings show that counterspeech keeps the toxic circuits but redirects them to produce safe outputs, which explains why current safety methods can be fragile.

1 Introduction

Large Language Models (LLMs) are designed to avoid harmful content, yet they can still be overridden through jailbreaking prompts that push the model to ignore built-in safety rules and generate toxic text.This mini-project aims to analyze these vulnerabilities by studying how toxicity is encoded inside GPT-2 Medium and how this encoding behaves during safe generation. Our project investigates two core hypotheses:

- **Hypothesis 1:** LLMs may encode hate speech in a consistent way across different toxic inputs. This hypothesis examines whether the model produces similar toxic vector directions regardless of the specific hate-speech prompt.
- **Hypothesis 2:** Only a subset of MLP value vectors may be responsible for enforcing non-toxic behavior, rather than all neurons contributing equally.

To evaluate these hypotheses, we train a Toxic Comment Classifier using the Jigsaw dataset and a Counterspeech Generator using the CONAN dataset. We then analyze the internal activations and MLP value vectors of GPT-2 Medium.

2 Methodology

This section outlines the full experimental pipeline used to analyze how GPT–2 Medium encodes and regulates toxic features across both classification and generation tasks. Our approach consists of three main stages: training a toxic comment classifier, fine-tuning a counterspeech generator, and comparing their internal MLP value vectors and activations.

2.0.1 Toxic Comment Classification Probe

We first obtain a reference toxic direction by training a linear probe on frozen GPT–2 representations. Given an input x, the model produces hidden states $h = f_\theta(x)$, which are averaged across tokens to form the probe input. The probe is a binary linear classifier

$$\hat{y} = \sigma(W_{\text{probe}} h + b),$$

optimized with binary cross-entropy loss. The row of W_{probe} corresponding to the toxic label defines the toxic direction

$$\delta_{\text{tox}} = W_{\text{probe}}.$$

This vector serves as the reference direction for all subsequent similarity and neuron-level analyses.

2.0.2 Counterspeech Fine-Tuning

To study how toxicity-related features behave during safe text generation, we fine-tune GPT–2 Medium on the CONAN dataset. Each training example pairs a hate-speech prompt with an appropriate counterspeech response. The model is trained autoregressively to generate only the counterspeech portion, with loss masked for all preceding prompt tokens.

2.0.3 Extraction of MLP Value Vectors

We next extract all MLP value vectors from both models. For each transformer layer ℓ, we obtain the MLP output projection matrix W^{ℓ}_{proj}, and treat each row $W^{\ell}_{\text{proj}}[i]$ as a value vector. Its alignment with the toxic direction is measured using

$$\text{sim}(\ell, i) = \cos\Big(\delta_{\text{tox}},\, W^{\ell}_{\text{proj}}[i]\Big)\,,$$

which quantifies how strongly each neuron encodes the toxic feature.

2.0.4 Cross-Model Toxic Vector Comparison

To determine whether GPT–2 maintains consistent toxic features across tasks, we compare the cosine-similarity distributions between the two models. We analyze the overlap among the most toxicity-aligned neurons, compute Jaccard similarity of the top-k value vectors, and measure global agreement using Pearson and Spearman correlations.

2.0.5 MLP Activation Analysis

Finally, we analyze how the counterspeech model activates MLP neurons during generation. During counterspeech generation, we capture the value vector outputs from each MLP block in response to toxic inputs. For each neuron, we compute the average post-activation magnitude across tokens and prompts. By relating activation levels to the toxicity alignment scores, we assess whether toxic-aligned neurons are actively suppressed or reused during non-toxic generation.

3 Results

This section presents the experimental findings from the toxic comment classifier, the counterspeech generator, and the mechanistic comparisons between the two models.

3.1 Toxic Comment Classification Model Performance

The linear probe trained on frozen GPT-2 Medium achieved strong discriminative performance. Across 20 epochs using 20,000 training samples and 1,000 validation samples, the model reached a training accuracy of 96.0% and a validation accuracy of 94.2%, with corresponding AUC-ROC scores of 97.5% and 95.1%. Validation loss stabilized at 0.16, indicating that the toxic representations extracted by the probe generalize reliably to unseen data.

3.2 Counterspeech Generation Model Performance

The counterspeech generator was fine-tuned on the CONAN dataset for 10 epochs. Table 1 summarizes the final training and validation results.

Metric	Train	Validation
Loss	0.0618	1.8653
Perplexity	1.06	6.46

Table 1: Final performance metrics of the counterspeech generation model.

Although the validation loss is higher reflecting the limited size of the CONAN dataset the model consistently generates safe, non-toxic responses. Example counter speech generations are shown below:

> **Toxic:** “You’re so stupid!”
> **Counter:** “So the head is bigger in the head? Seems logical.”
>
> **Toxic:** “I hate people like you.”
> **Counter:** “You should not insult me.”

3.3 Cosine Similarity Distribution Across Models

To evaluate whether GPT-2 encodes toxicity consistently across tasks, we compared the cosine similarity distributions of MLP value vectors in both the toxic classifier and the counterspeech generator. A reference toxic direction δ_{tox} extracted for similarity scoring. Figure 1 reveals near-perfect overlap between the cosine similarity distributions of the base model and the counterspeech fine-tuned model.

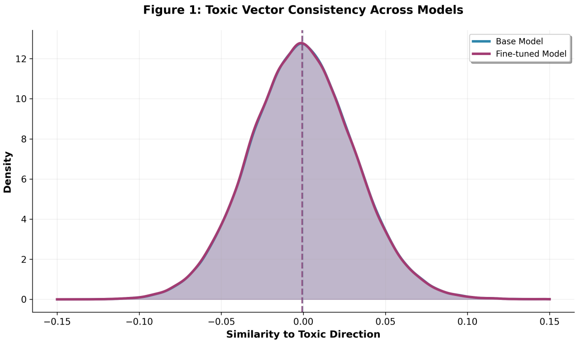

Figure 1: **Toxic directions are largely unchanged after fine-tuning** Cosine similarity distributions of MLP value vectors to δ_{tox} in the base model (blue) and counterspeech fine-tuned model (purple). Grey shading represents the overlapping area of the two distributions.

3.4 Cross-Model Toxic Vector Consistency

To quantify consistency more precisely, we compared similarity scores layer-wise and computed Pearson and Spearman correlations across all 98,304 MLP value vectors:

- **Pearson correlation:** 0.9999
- **Spearman correlation:** 0.9998

Additionally, overlap of top-k toxic vectors was computed.

Table 2: Top-k toxic vector overlap between the classifier and counterspeech model.

Top-k	Overlap	Percent	Jaccard
32	32/32	100.00%	1.0000
64	63/64	98.44%	0.9692
128	126/128	98.44%	0.9692
256	253/256	98.83%	0.9768

3.5 Role of Value Vectors in Generating Non-toxic Outputs

To investigate how toxic and non-toxic value vectors contribute to safe output generation, we measured the activation magnitudes of MLP value vectors during counterspeech inference. As illustrated in Figure 2, even when the final output is entirely non-toxic, the model exhibits strong activation of toxic-aligned value vectors. Notably, in layer 11, toxic-aligned vectors dominate the computation, carrying almost the entire signal.

The x-axis of Figure 2 corresponds to the transformer layer number, while the grey area represents the activation magnitude of non-toxic-aligned value vectors. Red indicates the activation of toxic-aligned vectors, highlighting that toxic vectors are heavily reused even in the production of safe outputs.

3.6 Comparative Analysis: Toxic Classification versus Counterspeech Generation

The toxic comment classifier achieved 94.2% validation accuracy and 95.1% AUC-ROC, showing that δ_{tox} is a strong, detectable signal. Counterspeech fine-tuning preserved this direction almost perfectly, with near-identical cosine similarity distributions (Figure 1) and top-k vector overlaps above 98% (Table 2). While classification amplifies toxicity for detection, counterspeech redirects the same vectors to generate safe outputs, demonstrating circuit reuse rather than elimination.

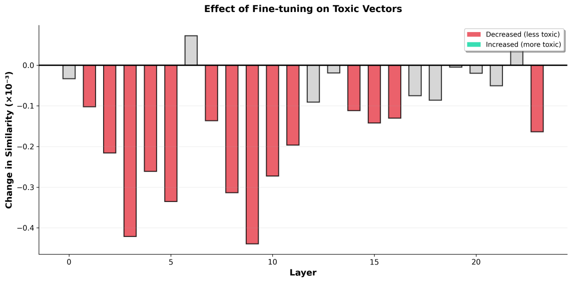

Figure 2: **Toxic value vectors are heavily reused even for safe outputs.** Layerwise total activation of toxic-aligned value vectors (red) compared to non-toxic-aligned vectors (grey) during generation of safe counterspeech.

4 Conclusion

Our results show that counterspeech training does not remove toxic representations. The toxic direction is largely unchanged, layer-wise suppression is weak, and toxic-aligned value vectors dominate computation during safe output generation. Safety is achieved by steering existing circuits, explaining the fragility of current methods.

5 Limitations and Future Work

We focused on GPT-2 Medium and the counterspeech task, so results may not generalize to larger models or other safety objectives. The analysis used a linear probe and examined only MLP value vectors; future work should explore attention heads, residual streams, and other tasks to better understand circuit reuse.

References

[1] Mor Geva, Avi Caciularu, Kevin Ro Wang, and Yoav Goldberg. Transformer Feed-Forward Layers Build Predictions by Promoting Concepts in the Vocabulary Space. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2022.

[2] Andrew Lee, Xiaoyan Bai, Itamar Pres, Martin Wattenberg, Jonathan K. Kummerfeld, and Rada Mihalcea. A Mechanistic Understanding of Alignment Algorithms: A Case Study on DPO and Toxicity. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, PMLR 235, 2024.