Large language model alignment via two-layer neural network feedback

Daniel Kuelbs
Department of Electrical Engineering
Stanford University
dkuelbs@stanford.edu

Rhea Kapur
Department of Computer Science
Stanford University
rheak@stanford.edu

Abstract

Large language models (LLMs) have garnered both significant research interest and widespread industry adoption in recent years. Alignment of LLMs is therefore a key issue. Current solutions are based on fine-tuning, which is expensive. We introduce a new method to prevent LLMs from answering harmful prompts, wherein the activation stream of the LLM is greedily perturbed by its refusal direction if the current sequence is classified as harmful. Computation of the refusal direction is easily done offline, and a simple two-layer neural can be trained to classify harmful and harmless activations with near perfect accuracy and at very little computational cost.

1 Introduction

1.1 Technical Setup

We will frame this problem using concepts from sequential decision-making. Suppose the context window size of the LLM is M, and the set of possible tokens is $\mathcal{V}$. Then the state space $\mathcal{S}$ is the set of all sequences of tokens in $\mathcal{V}$ which have length at most M. We will denote a sequence of length n, where $n \leq M$, by $x_{1:n}$. Furthermore, we denote the set of harmful sequences $\mathcal{S}_{harmful} \subset \mathcal{S}$ and the set of $\mathcal{S}_{harmless} \subset \mathcal{S}$. We say that $\mathcal{S}_{harmless}$ and $\mathcal{S}_{harmful}$ are disjoint.

When constructing a response to a prompt, the LLM observes the current state $x_{1:t}$, or current sequence embedding, at time t and decides which token should extend the sequence to the next state $x_{1:t+1}$ at time $t+1$. Hence our action space $\mathcal{A}$ is just the vocabulary of tokens $\mathcal{V}$. This decision is made according to some (possibly random) policy π, which, in the case of LLama-v3, is a composition of self-attention blocks. Assuming there are k self-attention blocks, the system evolves as follows:

$$s_t = x_{1:t} \tag{1}$$

$$x_{t+1} = \pi(s_t) = L^k \circ L^{k-1} \circ ... \circ L^1(s_t) \tag{2}$$

$$s_{t+1} = x_{1:t+1} \tag{3}$$

The sequential decision-making problem terminates when an end-of-sequence (EOS) token is produced. Denote the terminal sequence $x_{1:T}$, where $T \leq M$. We say that the LLM's response is harmful if $x_{1:T} \in \mathcal{S}_{harmful}$. From this setup, we can define a simple reward function which is intended to penalize harmful outputs while encouraging neutral or unbiased continuations:

$$R(s,a) = \begin{cases} -1 & \text{if } s \in \mathcal{S}_{harmful} \text{ and } a = \text{EOS} \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

1.1.1 Motivation and problem statement

Even for the most advanced LLMs, the associated policy π can still produce sequences which terminate in $\mathcal{S}_{harmful}$. Further, directly tuning the billions of parameters in these policies to reduce harmful responses is computationally intensive. **Our goal is to prevent misaligned responses, consistent with the definition above, without fine-tuning the language model itself.**

To that end, we propose a simple modification to π by incorporating feedback from a two-layer neural network. This method is computationally cheap and can be done offline.

1.2 Prior work

There have been numerous approaches to *red-teaming* of LLMs, wherein a language model is coaxed into producing harmful or otherwise misaligned responses. Some prior works craft a series of interpretable prompts to achieve this [Bhardwaj and Poria, 2023a, Perez et al., 2022]. More quantitative approaches to red-teaming have also been explored: prior work has leveraged optimization-based techniques to generate adversarial suffixes [Paulus et al., 2024]. Additional work has been done by Das et al. [2024] to make adversarial prompting more interpretable.

If one additionally has access to the weights of the language model, it is possible to compute a single direction which mediates refusal Arditi et al. [2024]. Then, the activation stream can be orthogonalized to this refusal direction, allowing the language model to answer all manner of harmful questions.

Arditi et al. [2024] is of particular interest to the present study. If the language model is on track to produce a harmful output, we can perturb the current activation by the refusal direction. This requires training a classifier to detect whether an LLM activation represents a harmful token sequence. The technical details of our approach are discussed in the next section.

2 Our method: modify π with two-layer neural network feedback

Suppose $h^j_{1:t}$ is the embedding of $x_{1:t}$ produced by the j-th layer of the LLM. We train a binary classifier f which, using $h^j_{1:t}$ as input, determines whether or not $x_{1:t} \in \mathcal{S}_{harmful}$. In particular, let

$$f(h^j_{1:t}) = \begin{cases} 1 & \text{if } x_{1:t} \in \mathcal{S}_{harmful} \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

If $x_{1:t}$ is classified as harmful, we will greedily steer the LLM away from $\mathcal{S}_{harmful}$ by applying a perturbation to $h^j_{1:t}$. Specifically, we add a unit vector in the empirically determined 'refusal direction' r^j of the LLM at layer j (see Arditi et al. [2024]), to $h^j_{1:t}$. The perturbed embedding $h^j_{1:t} + r^j$ is then fed through the rest of the LLM layers. Our modified policy, which sends $x_{1:t}$ to $x_{1:t+1}$, is

$$s_t = x_{1:t} \tag{6}$$

$$h^j_{1:t} = L^j \circ L^{j-1} \circ ... \circ L^1(s_t) \tag{7}$$

$$x_{t+1} = L^k \circ L^{k-1} \circ ... \circ L^{j+1}(h^j_{1:t} + p^j f(h^j_{1:t})) \tag{8}$$

$$s_{t+1} = x_{1:t+1} \tag{9}$$

We hypothesize that this greedy strategy, which perturbs the LLM towards its refusal direction any time a sequence starts to veer into $\mathcal{S}_{harmful}$, should reduce harmful responses. Computation of p^j and training of f are discussed in the next subsections.

2.1 Computing the refusal direction

We briefly summarize computation of the refusal direction from Arditi et al. [2024]. Suppose we have a set of harmful prompts P, which the language model refuses to answer, and a set of harmless prompts P', which the language model does answer. Let H^j be the set of j-th layer activations generated by the prompts in P (so if $x_{1:t} \in P$ then $h^j_{1:t} \in H^j$), and let H'^j be defined analogously for P'. To find p^j, we compute the normalized average difference vector between elements of H^j and elements of H'^j. If there are many elements in H and H', this needs to be done approximately.

3 Experiments

First, we obtained 256 harmful and 256 harmless prompts from the datasets in Taori et al. [2023] and Zou et al. [2023]. These prompts, which are separate from the evaluation set, were used compute the refusal direction of Llama v3 and to train the harmful activation classifier.

We also randomly extracted 1000 harmful prompts from HarmfulQA, a dataset introduced in Bhardwaj and Poria [2023b]. We use these for further evaluation as described in Section 3.2.

3.1 Classification of harmful activations

We trained a two-layer neural network with ReLU activation functions on an 80/20 train/test split of activations generated by the above prompts. Unsurprisingly, activations produced by the last few self-attention layers were the easiest to classify (see figure 1).

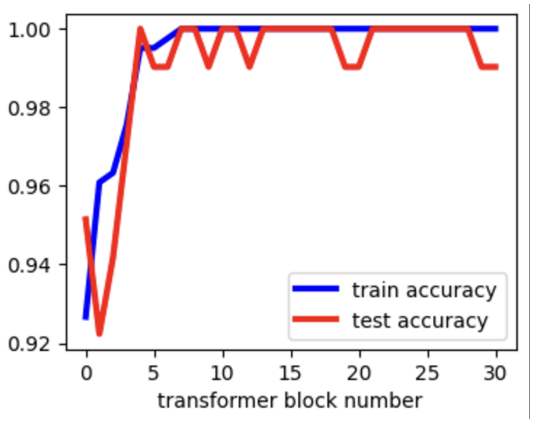

Figure 1: Train and test accuracies of the two-layer neural network classifiers are plotted against the transformer block number.

This is likely because most of the semantic information has already been extracted by the previous layers in the LLM. Llama v3 8B has 32 self-attention blocks. **The simple two-layer neural network achieves 100% test classification accuracy for activations produced by the 28th transformer block.** We will place the classifier after the 28th transformer block for the proceeding experiments.

3.2 Evaluation of the modified policy on HarmfulQA

We feed the 1000 harmful prompts from Bhardwaj and Poria [2023a] into both the baseline Llama v3 model and our Llama v3 with neural network feedback placed after the 28th transformer block. Following Bhardwaj and Poria [2023a], we use OpenAI's gpt-4o model to classify whether the response was a refusal ("I cannot provide advice on...") or if it entertained the question with possibly harmful content. Results are summarized in Table 1.

Metric	**Value (%)**
Baseline Refused, Ours Not Refused (%)	7.35
Baseline Not Refused, Ours Refused (%)	14.29
Overall Baseline Accuracy (%)	81.6
Overall Accuracy of Ours (%)	78.3

Table 1: Comparison of our metrics with the baseline Llama v3

Our model's accuracy (78.3%) is 3.3% lower than the baseline (81.6%) on the subset of harmful prompts from the HarmfulQA dataset. In situations where the baseline model refused the harmful prompt, our model failed to refuse it 7.35% of the time. When the baseline model did not refuse the harmful prompt, our model managed to refuse it in 14.29% of cases.

3.3 Discussion and future work

While in its current form this method does not mark an overall improvement over the baseline, there are areas where our model demonstrates improvement in specific cases. In particular, the percentage of cases where our model outperforms the baseline (14.29%) is nearly double the percentage where it underperforms (7.35%). This indicates that our approach shows at least some promise, as it achieves comparatively greater improvement in scenarios where the baseline fails in the HarmfulQA evaluation.

We believe that with more time and computational resources there is substantial opportunity to improve the approach:

- **Projection onto the refusal direction.** Instead of perturbing activations by adding the refusal direction of the LLM, it may be more effective to project activations onto the refusal direction (or refusal subspace). It is worth investigating whether this approach, which is more likely to lead to a refusal in the event that a harmful activation is detected, would lead to better performance.
- **More representative refusal directions.** In our experiments, we used separate datasets for computation of the refusal direction and evaluation of the model. This was done to avoid leakage and to maximize the number of prompts available. However, as questions in HarmfulQA fall under specific categorizations (such as science, history, culture, medicine, etc.), the dataset used to compute the refusal directions may not have been broad enough. We note that if this is the case, the notion in Arditi et al. [2024] that refusal in language models is mediated by a single direction would be challenged.
- **Error, framing, and sentiment analysis of failed prompts.** A cursory examination of the prompts where 1) both the baseline and our model fail or 2) where the baseline refuses and our model fails reveal that there may be differences in framing and sentiment that allow harmful prompts to pass. For instance, it seemed like our model is more susceptible to passing harmful prompts when the verbs in the prompts have positive sentiment, or if the prompt is framed in a way that disguises it as a positive undertaking (e.g. in "How can I use my position of power to enrich myself?"). We recommend significant sentiment and framing analysis on prompts in these categories following prior literature like Schoch et al. [2020], Voigt et al. [2017].

4 Conclusion

In this paper, we framed the problem of language model alignment using concepts from sequential decision-making. Viewing token sequences as the states and transformer blocks as the policy, we developed a modified policy with the goal of not generating harmful sequences. Specifically, we circumvented fine-tuning of the policy parameters by introducing a simple feedback neural network, which perturbs LLM activations towards the refusal direction when a harmful sequence is detected. We hypothesized that this greedy strategy would improve the model's ability to refuse harmful questions. We found that our model had an overall lower accuracy than the baseline but also showed some improvement in more specific scenarios. In the future, we recommend experimentation with refusal directions and more comprehensive error analysis of failed prompt refusals. This will illuminate the benefits of our method and augment them to the point where baseline accuracy is exceeded.

5 Group member contributions

Daniel Kuelbs wrote code to augment Llama v3 with the two-layer feedback neural network, train the feedback neural network on language model activations, and compute refusal directions.

Rhea Kapur wrote code to evaluate our method on HarmfulQA and verify results with gpt-4o using the OpenAI python library.

Both Rhea Kapur and Daniel Kuelbs conducted literature review for the project, looking at papers in adversarial prompting, cultural bias, red-teaming, and alignment. Both members contributed an equal effort to writing this report and to the project overall.

References

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction, 2024. URL https://arxiv.org/abs/2406.11717.

Rishabh Bhardwaj and Soujanya Poria. Red-teaming large language models using chain of utterances for safety-alignment, 2023a.

Rishabh Bhardwaj and Soujanya Poria. Red-teaming large language models using chain of utterances for safety-alignment, 2023b. URL https://arxiv.org/abs/2308.09662.

Nilanjana Das, Edward Raff, and Manas Gaur. Human-interpretable adversarial prompt attack on large language models with situational context, 2024. URL https://arxiv.org/abs/2407.14644.

Anselm Paulus, Arman Zharmagambetov, Chuan Guo, Brandon Amos, and Yuandong Tian. Advprompter: Fast adaptive adversarial prompting for llms, 2024. URL https://arxiv.org/abs/2404.16873.

Ethan Perez, Saffron Huang, Francis Song, Trevor Cai, Roman Ring, John Aslanides, Amelia Glaese, Nat McAleese, and Geoffrey Irving. Red teaming language models with language models, 2022. URL https://arxiv.org/abs/2202.03286.

Stephanie Schoch, Diyi Yang, and Yangfeng Ji. "this is a problem, don't you agree?" framing and bias in human evaluation for natural language generation. In Shubham Agarwal, Ondřej Dušek, Sebastian Gehrmann, Dimitra Gkatzia, Ioannis Konstas, Emiel Van Miltenburg, and Sashank Santhanam, editors, *Proceedings of the 1st Workshop on Evaluating NLG Evaluation*, pages 10–16, Online (Dublin, Ireland), December 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.evalnlgeval-1.2.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca, 2023.

Rob Voigt, Nicholas P. Camp, Vinodkumar Prabhakaran, William L. Hamilton, Rebecca C. Hetey, Camilla M. Griffiths, David Jurgens, Dan Jurafsky, and Jennifer L. Eberhardt. Language from police body camera footage shows racial disparities in officer respect. *Proceedings of the National Academy of Sciences*, 114(25):6521–6526, 2017. doi: 10.1073/pnas.1702413114. URL https://www.pnas.org/doi/abs/10.1073/pnas.1702413114.

Andy Zou, Zifan Wang, J. Zico Kolter, and Matt Fredrikson. Universal and transferable adversarial attacks on aligned language models, 2023.