Preliminary Steps Toward Investigating the "Smearing" Hypothesis for Layer Normalizing in a 1-Layer SoLU Model[1]

Mateusz Bagiński*
Independent

Alana Xiang*
Stanford University

Kunvar Thaman*
Independent

Rohan Gupta*
IIT Bombay

j1ng3r
Stanford University

Neel Nanda, Esben Kran, Fazl Barez

Abstract

SoLU activation functions have been shown to make large language models more interpretable, incentivizing alignment of a fraction of features with the standard basis. However, this happens at the cost of suppression of other features. We investigate this problem using experiments suggested in Nanda's 2023 work "200 Concrete Open Problems in Mechanistic Interpretability".

We conduct three main experiments. 1, We investigate the layernorm scale factor changes on a variety of input prompts; 2, We investigate the logit effects of neuron ablations on neurons with relatively low activation; 3, Also using ablations, we attempt to find tokens where "the direct logit attribution (DLA) of the MLP layer is high, but no single neuron is high."

** Denotes equal contribution.*

Keywords:Mechanistic interpretability, AI safety

1. Introduction

Comprehensive understanding of the neural networks' internals in terms of decomposable features is a major goal for the field of mechanistic interpretability (Olah et al., 2020). One thing that could substantially accelerate progress on that front would be modifications to the current architectures that make them easier to interpret, while incurring little to no performance penalty.

One way to make neural networks easier to interpret is to use activation functions that incentivize the features learned by the model to align with neurons (in other words, make

[1] Research conducted at the Apart Research Alignment Jam #10 (Interpretability 3.0), 2023 (see https://alignmentjam.com/jam/interpretability)

the standard basis interpretable; Elhage et al., 2021). Superlinear activation functions (Elhage et al., 2022a) are functions that increase the activation of a neuron relative to its pre-activation value at a greater than linear rate. This amplifies activations with high relative values, while at the same time suppressing activations with small relative values. Since this selection occurs at the neuron level, it was suggested that an activation function of that kind may incentivize the model to learn feature representations that are aligned with neurons. At the same time, relative suppression of neurons with relatively low activations, may lower superposition.

One superlinear activation function is SoLU (Softmax Linear Unit) introduced by Elhage et al. (2022a), who trained GPT-style language models of several sizes using either SoLU or GELU (the current standard in LLMs) as their activation function. They found an increase in the fraction of MLP neurons that could be efficiently understood and explained by a human was higher in SoLU models, compared to GeLU models, from 30-40% to 40-60% (depending on the model size). Although merely changing the activation function decreased the model's performance, adding an additional LayerNorm just after the MLP restored the performance of the SoLU models to levels not significantly different from that of the GELU models of the same size.

However, SoLU incurred a different kind of cost in that while many neurons became interpretable, some (probably relatively less important) features became obscured or hidden. Since this did not harm performance, the model must have been able to use them despite their suppression. Thus, Elhage et al. (2022a) suggest that these features are being distributed (or "smeared out") across many neurons and recovered when necessary.

One candidate model component that may be responsible for recovering these features is LayerNorm, which includes scaling every neuron in the residual stream by a factor (typically denoted γ) specific to each neuron. This is a mechanism LayerNorm might use to amplify features in suppressed neurons.

Is it possible that transformers intentionally cultivate outlier feature dimensions as a mechanism to suppress the clutter that emerges from superposition and interference? Instead of eliminating, these dimensions may serve to dampen the noise. Could this be a strategy to effortlessly activate a 'noise suppression feature' in contexts prone to interference, rather than altering the complex process in which information is encoded in superposition?

Please note that this paper was written quickly in a Hackathon and may contain errors. Any code provided may be buggy, and our results have not been carefully checked. Nevertheless, we are excited to share the results of our fun team project. The code is available in the linked GitHub repo,[2] with sections copied and/or modified from Neel Nanda's excellent *Exploratory Analysis Demo* notebook.

2. Methods & Results

In this work, we performed several experiments on the actions of the LayerNorm in the one-layer SoLU model. Our experiments are based on instructions provided under

[2] https://github.com/firstuserhere/interp-hackathon-layernorm

Problem 4.39 in Neel Nanda's "200 Concrete Open Problems in Interpretability" sequence (Nanda, 2023).

$$y=\gamma(x-\mathrm{E}[x])/(\mathrm{Std}[x]+\epsilon)+\beta$$

LayerNorm. (Ba et al., 2016)

1. As suggested in Nanda, we observe changes in the LayerNorm scaling factor given a variety of inputs. More specifically, we observe the change in the inverse of the standard deviation of the MLP-block LayerNorm layer's input tensor over GPT-generated sentences of various lengths, word repetitions ("BOS ice ice ice…") of various lengths, and overlapping random word strings of various lengths.
2. We observe changes from neuronal ablation on the performance of SoLU-1L on a variety of inputs. We attempt to use this method to identify neurons which seem to have surprisingly low activation given their high counterfactual impact on the final logit output.
3. We identify the MLP DLA over a variety of tokens, and make an attempt to find tokens where "the direct logit attribution (DLA) of the MLP layer is high, but no single neuron is high."

We used Neel Nanda's TransformerLens library (Nanda 2022) and the library's SoLU-1L model.

We generated prompts with Chat-GPT, using both the GPT-3 and GPT-4 models available on the `chat.openai.com` interface. The prompts used are available in the linked GitHub repository.

Part 1

We consider the effects of LayerNorm on a variety of inputs.

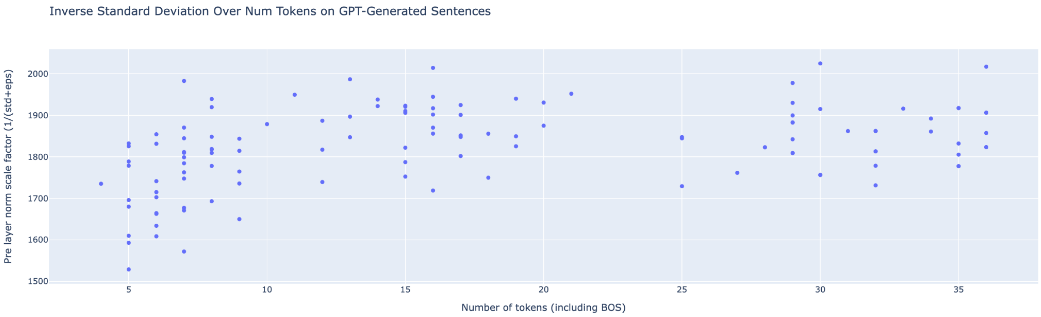

Figure 1. The LayerNorm scale factor seems to grow vaguely logarithmically with respect to the number of tokens when the model is fed GPT-generated sentences of varying lengths.

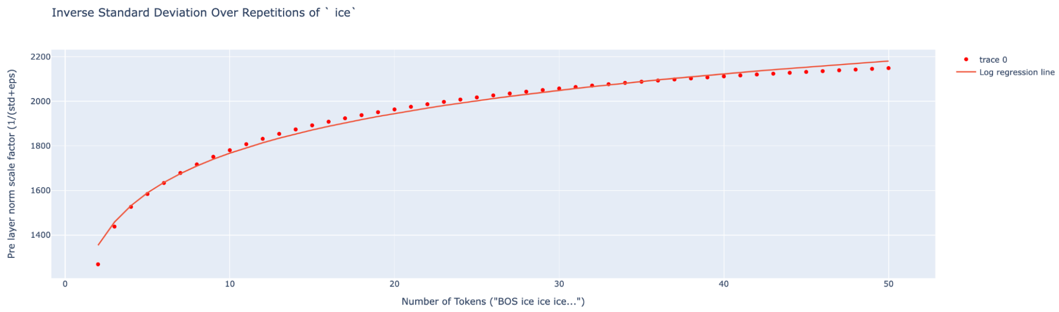

Figure 2. The LayerNorm scale factor seems to grow logarithmically with respect to the number of tokens when fed prompts of the form “BOS ice ice ice ice…”. The word “ice” was chosen due to nice tokenization properties.

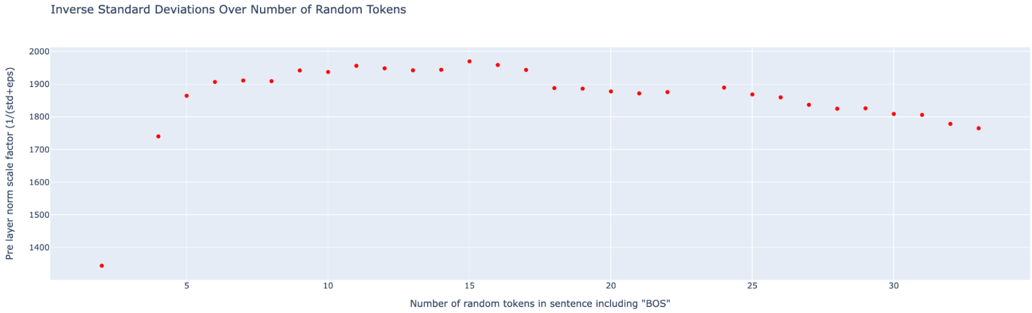

Figure 3. When fed substrings [:i] of a string of randomly generated words, it seems that ~15 tokens is the sweet spot. Above 25 tokens, we start seeing a steady decrease in the LayerNorm scaling factor, as opposed to what we may naively expect based on Figure 1 and Figure 2.

Part 2

Here, we looked for tokens where the neuron has (comparatively) low activation before the LayerNorm but high direct logit attribution after the LayerNorm. For each position in each prompt from a subset of those generated by GPT, we took the 10% of activations between the MLP and before the subsequent LayerNorm with the smallest values. Then we run the model again on each prompt, this time ablating the corresponding values after that LayerNorm and recorded how much the loss changed in response to this ablation. Since we investigated a one-layer model, the only path from the MLP to the logits is the direct one. Thus, using ablations should give the same result as using DLA.

The two scores were plotted on a scatter plot in order to see whether some distinct cluster with high ablation score formed. We found none. The experiment was repeated for all the activations but we did not find anything either.

Scores for the lowest 10%:

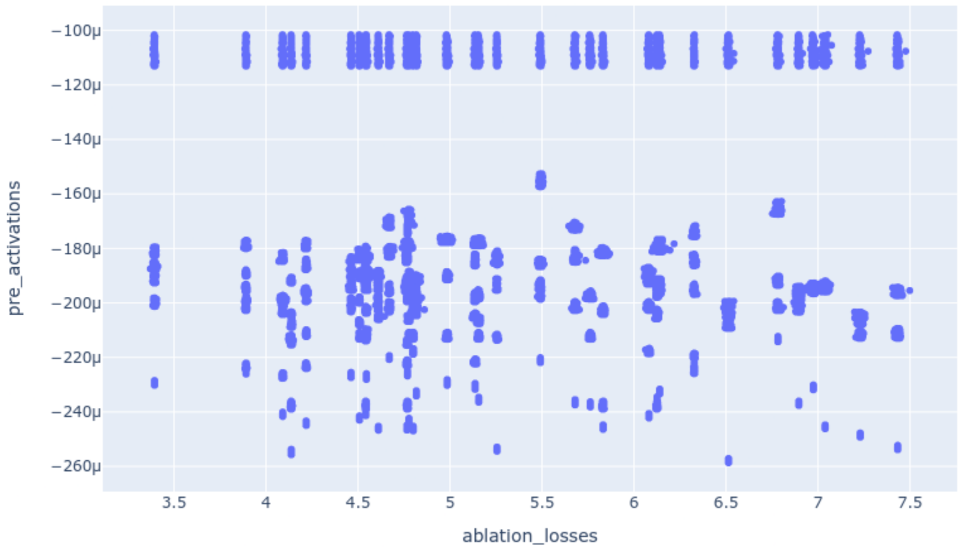

Figure 4. Pre activations vs ablation loss.

Scores for all activations:

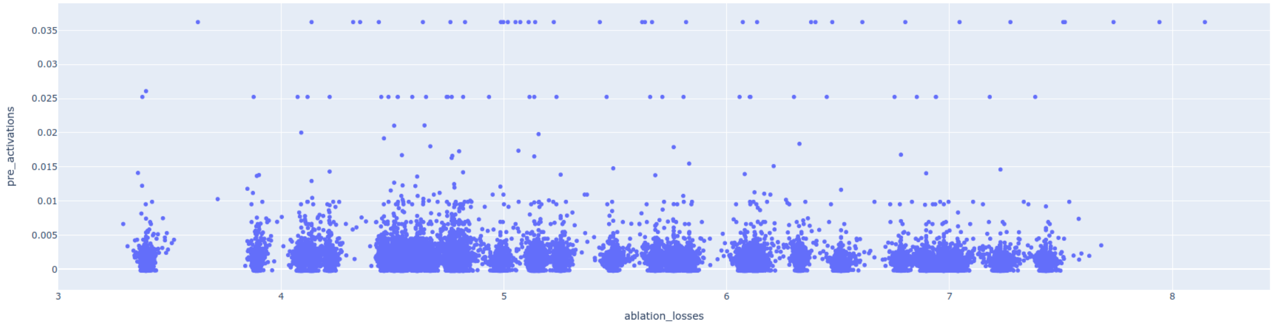

Figure 5. Pre activations vs ablations losses for all activations.

In this experiment, we looked for tokens in the post-MLP pre-LN activations where the contribution to logits is high, but no single neuron is high. For that purpose, we used the following to metric, computed for each token.

Ablation loss - how much ablating that position/token on that prompt increases loss.

Inverse outlier score - computed as 1 / (maximum_activation - mean_activation). The higher this score, the smaller the difference between the biggest activation at that position and the mean activation.

For each position, we multiplied ablation loss and inverse outlier score. Then, we tested whether positions where these scores are high are scaled by the LayerNorm to a greater extent than random positions. I.e., we calculated the ratio between these positions mean values after the LayerNorm and before the LayerNorm. However, no difference was found.

Part Three

We look for “special tokens” that seem to lead to MLP layers with high direct logit attribution without having neurons with high direct logit attribution.

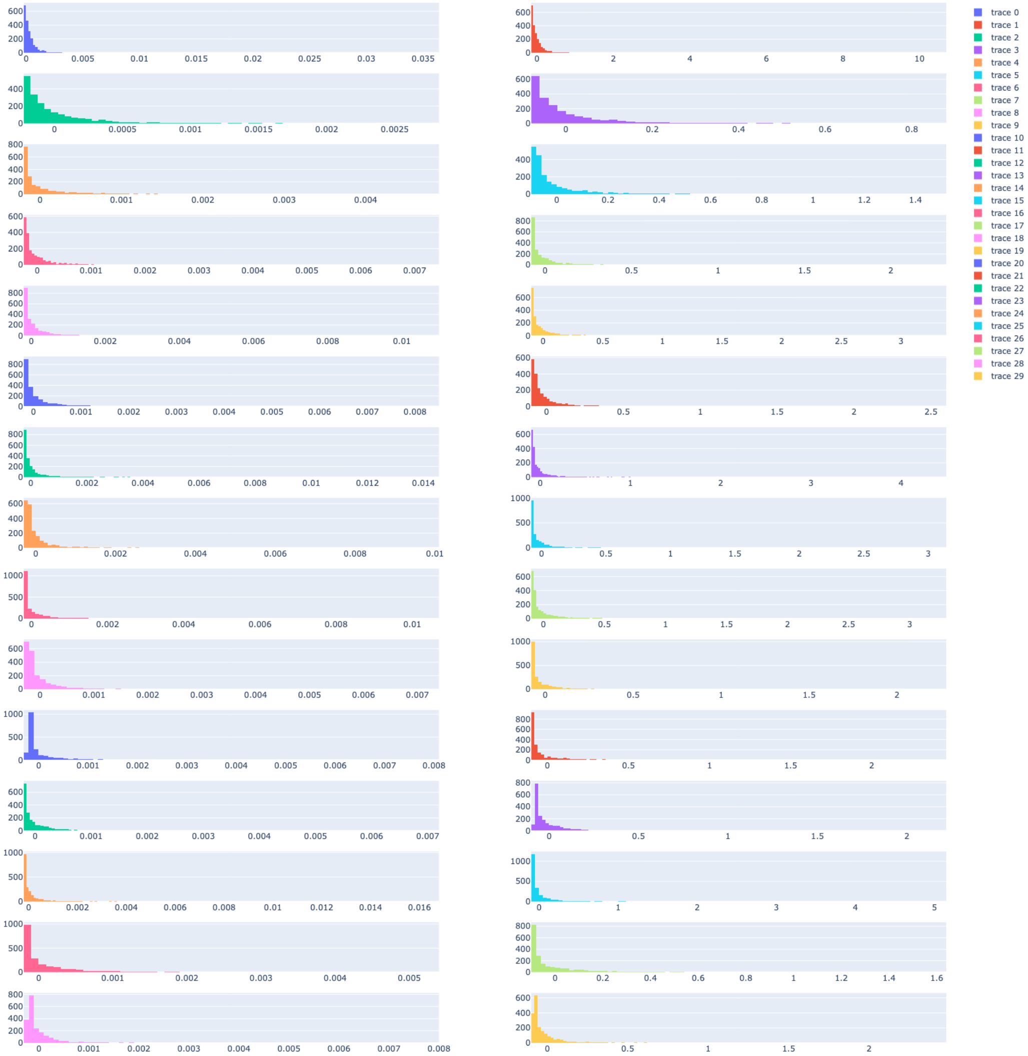

Figure 6. Distribution of MLP Direct Logit Attributions on different tokens.

3. Discussion and Conclusion

We have failed to find compelling evidence of the “smearing” hypothesis.

An obvious direction for future work is replicating our experiments on a broader set of prompts.

We consider the following to be open questions:

1. Why does the LayerNorm scaling factor decrease for long input tokens in the random word case but seemingly grow logarithmically in the repetition case?
2. Do our ablation experiments produce evidence of smuggling that we didn't recognize?

Ultimately, the question of whether the smearing hypothesis explains SoLU-1L's suspicious performance gain from LayerNorm remains unresolved by this paper. We are interested in pursuing more experiments to settle this question.

4. References

Elhage, et al., "A Mathematical Framework for Transformer Circuits", Transformer Circuits Thread, 2021.

Elhage, et al., "Softmax Linear Units", Transformer Circuits Thread, 2022a.

Elhage, et al., "Toy Models of Superposition", Transformer Circuits Thread, 2022b.

Olah, et al., "Zoom In: An Introduction to Circuits", Distill, 2020.

Goh, G., Cammarata, N., Voss, C., Carter, S., Petrov, M., Schubert, L., Radford, A., & Olah, C. (2021). Multimodal Neurons in Artificial Neural Networks. *Distill*, *6*(3), 10.23915/distill.00030. https://doi.org/10.23915/distill.00030

Lindner, D., Kramár, J., Rahtz, M., McGrath, T., & Mikulik, V. (2023). *Tracr: Compiled Transformers as a Laboratory for Interpretability* (arXiv:2301.05062). arXiv. http://arxiv.org/abs/2301.05062

Olah, C., Cammarata, N., Schubert, L., Goh, G., Petrov, M., & Carter, S. (2020). Zoom In: An Introduction to Circuits. *Distill*, *5*(3), 10.23915/distill.00024.001. https://doi.org/10.23915/distill.00024.001

Weiss, G., Goldberg, Y., & Yahav, E. (2021). *Thinking Like Transformers* (arXiv:2106.06981). arXiv. http://arxiv.org/abs/2106.06981

Nanda, N. (2023, January 3). *200 cop in MI: Exploring polysemanticity and Superposition - AI alignment forum*. AI Alignment Forum. https://www.alignmentforum.org/s/yivyHaCAmMJ3CqSyj/p/o6ptPu7arZrqRCxyz

Nanda, N. (2022). "TransformerLens." GitHub. https://github.com/neelnanda-io/TransformerLens.