# Interactive Layerscope[1]

**Víctor Levoso** **Alejandro González** **Chris Lonsberry**

## Abstract

We expand Neel Nanda's Interactive Neuroscope to view an entire layer.

*Keywords: Mechanistic interpretability, ML safety*

## 1. Introduction

Looking at Neel Nanda's Interactive Neuroscope, we were stymied by the question of which neuron we ought to try to look at. It seemed potentially useful to be able to quickly map the activations of every neuron in the layer, particularly for smaller models with manageable numbers of neurons. To that end, we build a new version of the Neuroscope which generates a graphical representation of the entire layer. Figure 1 below shows the UI for the layercope, hosted at https://969dd6aa-d0ee-4fb0.gradio.live/[2].

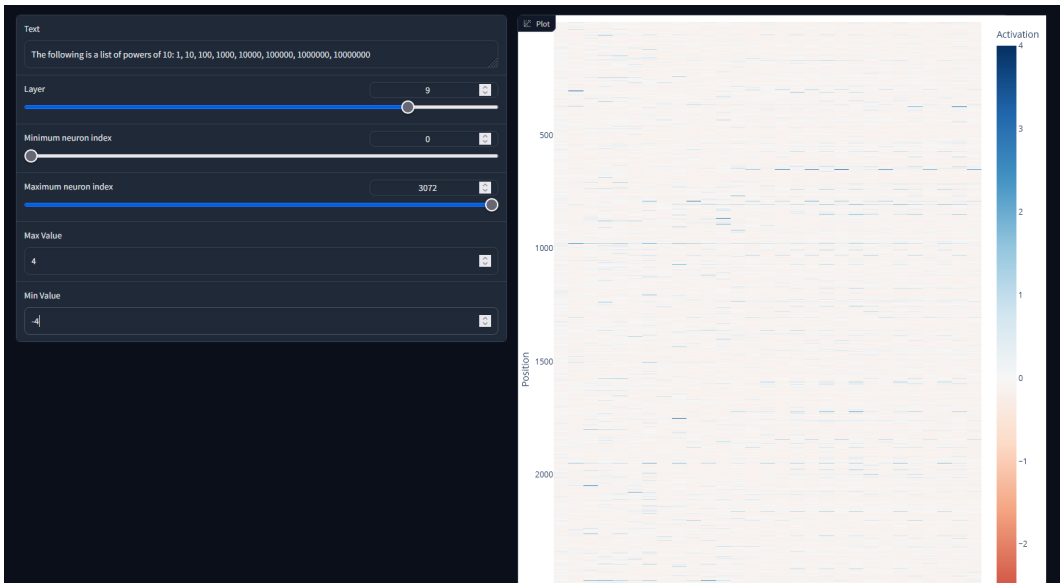


*Figure 1 – Interactive Layerscope*

[1] Research conducted at the Apart Research Alignment Jam #4 (Mechanistic Interpretability), 2023 (see https://itch.io/jam/mechint)

[2] Because the model is running in Colab, the site will only stay online until the Colab environment times out: about 20 hours after the submission deadline.

In figure 2 below, we show the layerscope image for layer 7, generated using the default text in Nanda's Neuroscope: "The following is a list of powers of 10: 1, 10, 100, 1000, 10000, 100000, 1000000, 10000000".

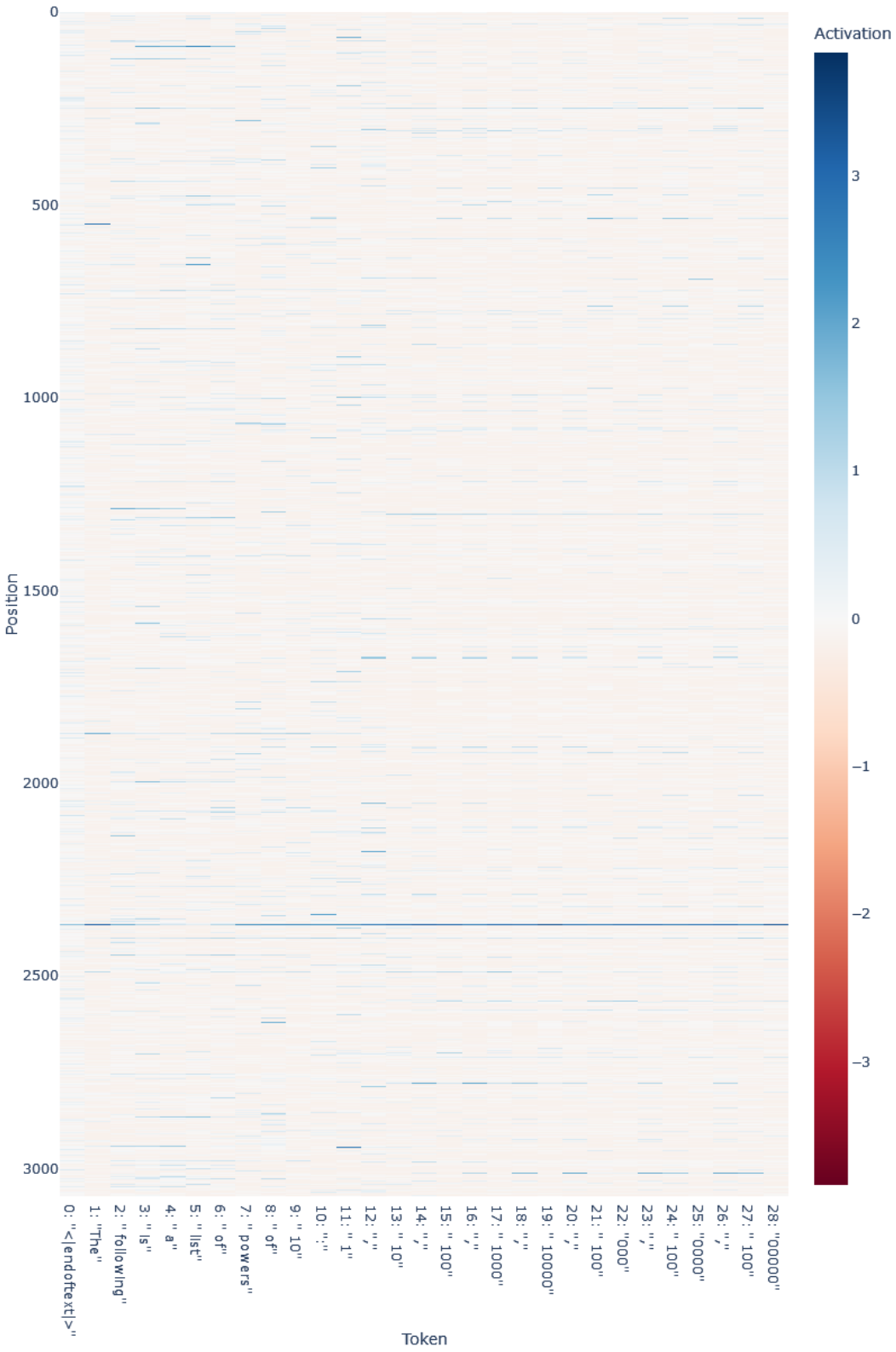

*Figure 2 – Neuroscope image for layer 7, GPT2-small, input: "The following is a list of powers of 10: 1, 10, 100, 1000, 10000, 100000, 1000000, 10000000"*

We analyze more prompts with this tool and identify some interesting patterns and possible avenues for further research.

## 2. Results

The main advantage of using the full layer map is that it can quickly identify which neurons are most activated in a given layer for the input text. An interesting approach is to cycle through the layers and watch structure emerge. For example, the comma tokens seem to activate across many neurons in many layers and the pattern appears to grow stronger in the later layers.

If we maintain the general structure of the default input text (i.e. a string of text followed by a series of numbers), one neuron becomes prominent: 7/2367[3]. It also seems to activate strongly on almost every token, beginning to strengthen in activation after 7 or 8 tokens. Looking up 7/2367 on neuroscope.io, we find that it is indeed unusual. It seems to activate strongly on unusual characters or unintelligible text. Unfortunately, we did not come any closer to understanding this neuron during the course of the hackathon.

The phenomenon of neurons that activate strongly on many tokens seems to occur with some frequency throughout the layers. The first three sentences of the introduction to this paper activate strongly on every token in the following neurons: 6/437, 10/900, 1690, 1793, 1973; and 11/611.

This raises an interesting question for further research: should we be interested in neurons that activate strongly on many tokens or should we focus instead on neurons which only activate strongly on one or two tokens? At present, we lean toward the idea that we should try to explain the neurons that only activate strongly on one or two tokens. First, many of the neurons that activate strongly on many tokens have unusual maximally activating strings in neuroscope.io. Secondly, on a more principled note, when a neuron activates strongly on many tokens it will be difficult to identify specific tokens that contribute to the high activation.

While looking through random neurons, we discovered 3/2032, which activates strongly on legal words related to violent crime, with a strong preference for the word "battery". We tested it against various prompts that refer to electrical batteries and it does not activate on those.

## 3. Discussion and Conclusion

It seems feasible to us that looking at a full layer as we have done could yield interesting avenues of investigation. To that end, we think it might be worthwhile to enhance the

[3] We will represent neurons as <layer>/<neuron>. Thus the above example is neuron 2367 in layer 7.

feature set beyond what we could do in this hackathon. We believe the following enhancements could provide additional value:

1. compile all the image layers into an animated image (e.g. gif) and
2. cache recent prompts and images for review, and
3. plot activations on the residual steam before the MLP.