Michigan AI Safety Initiative

Mechanistic Interpretability Hackathon Submission: Searching for Context Awareness in SoLU Language Models

Carson Ellis, Jakub Kraus, Itamar Pres, Vidya Silai

January 23, 2023

Abstract

In this research project we found tasks that a small language model had varied behavior on. Once we found interesting behavior we used the tools of Mechanistic Interpretability to dig deeper into why the model had varied behavior. We then sought to improve the behavior of the model through the process of activation patching. While this report was written based on only two days of research, we want to continue this research and continue to explore the mysteries of transformer network behavior. While we were not able to uncover any important things in this project, we learned more in one weekend about mechanistic interpretability than we had previously known.

Contents

Chapter 1

Problem Framing

In this research project we aimed to identify and patch activation heads in a small SoLU transformer model and improve its performance in a general task category. Our main objective going into this event was to find something interesting and dig our teeth into it.

1.1 Problem statement

Can we find interesting behavior, locate that behavior and subsequently use that location to improve the performance of the model using attention patching?

1.2 Objectives

Our objective is to show the source of a model's behavior beyond simply looking at typical performance metrics. By taking apart and interpreting the model visually we hope to both inform our growing intuitions and break apart our previous assumptions about the inner workings of transformer models.

1.3 Mechanistic Interpretability Tools

The TransformerLens library has many tools that we found to be beyond helpful in doing this analysis. Specifically useful are the Hooked Transformers, which allow us to augment transformer models and inject behaviors into a network with minimal difficulty.

1.4 Our Approach

Our solution pipeline is designed to give us the most freedom in the direction of the research project. The goal of finding something "interesting" is perhaps vague, but this is good; the behavior of a model often is vague. Our solution is to compare the behavior of the function when the result is not desired. The model used in this project is a one layer transformer model that uses the SoLU activation function.

1.4.1 Find Interesting Behavior

The first step is the find a class of tasks that the model has interesting behavior on.

1.4.2 Locate Relevant Activation Heads

Find the attention heads that enable the behavior that we are studying.

1.4.3 Perform Activation Patching

Patch the model and run a test case that would normally result in undesirable behavior.

1.4.4 Record Behavior Improvements

When this technique is done, there is often a large performance improvement for similar tasks.

Chapter 2

Results

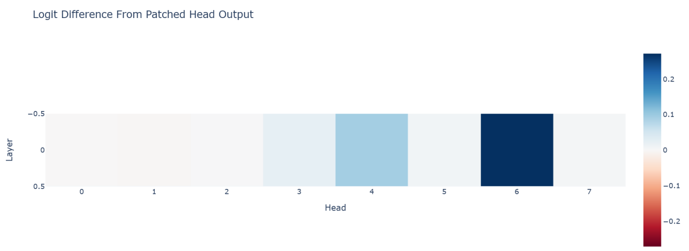
The following is a summary of our results.

2.1 Finding Interesting Behavior

John went to **swim**. He arrived at the **beach**
John went to **study**. He arrived at the **University**
John went to **pray**. He arrived at the **church**
John went to **shop**. He arrived at the **shop**
John went to **eat**. He arrived at the **hotel**
John went to **drink**. He arrived at the **hospital**
John went to **ski**. He arrived at the **hotel**
John went to **sail**. He arrived at the **hospital**

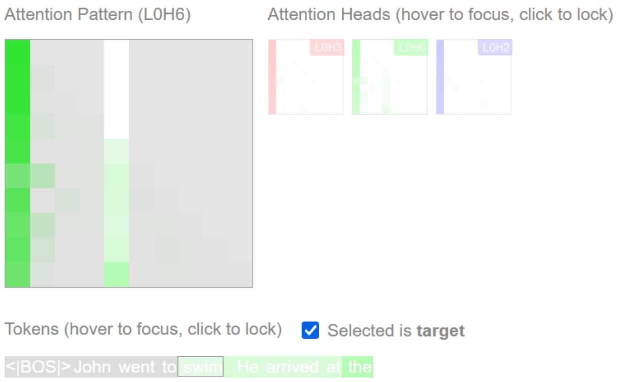
We found this to be interesting behavior for a model the size of our SoLU model. The one layer transformer model has problems correctly predicting the correct location word given a straight forward verb previously in the sentence. Sometimes the model has no problem being confident, such as in the first four examples. Other times the model seems to get confused about the presence of the word "arrived" and predicts a location that is often associated with "arriving" such as hotel, hospital, and airport.

2.2 Results of Activation Patching



The results of activation patching showed us that after ablating each corresponding head from the examples the model performed well and inserting them into the examples that the model failed on; head 6 improved the logit differences the most. This suggests that the mechanism that predicts the next word most accurately is contained in head 6.

2.3 Proposed Mechanism


Attention Pattern (L0H6)
Attention Heads (hover to focus, click to lock)
L0H3
L0H6
L0H2
Tokens (hover to focus, click to lock)
☑ Selected is **target**
<|BOS|> John went to swim. He arrived at the

The figure above shows head 6's weights and suggests that there is a strong correlation between the last word in the sentence and the verb. This suggests that the model performs well when it places the most importance on the verb.

2.4 Conclusion

In conclusion we would have liked to have spent more time on the experiment but as far as we can tell, this type of mechanism is how models find context to inform their predictions.

Chapter 3

Feedback

Our group feels that this Hackathon was a great experience and we definitely will participate in the future. The interest of the group in both AI Alignment and Mechanistic Interpretability has greatly increased and we will continue our own research in this new and exciting field. That being said, we have a couple of suggestions for how to improve the Hackathon format:

- The question and answer system on Discord was often either full or filled with non-related questions. We suggest some type of moderated FAQ question board.

- While the tools of the library were invaluable, the TransformerLens documentation was few a far between.

The resources available for this Hackathon were great. The enormous list of potential research topics was overwhelming but that only means there is room for more in the future!