# EXPLORATIONS OF SELF-REPAIR IN LANGUAGE MODELS

**Cody Rushing**[*], **Neel Nanda**

## ABSTRACT

Prior interpretability research studying narrow distributions has preliminarily identified self-repair, a phenomena where if components in large language models are ablated, later components will change their behavior to compensate. Our work builds off this past literature, demonstrating that self-repair exists on a variety of models families and sizes when ablating individual attention heads on the full training distribution. We further show that on the full training distribution self-repair is imperfect, as the original direct effect of the head is not fully restored, and noisy, since the degree of self-repair varies significantly across different prompts (sometimes overcorrecting beyond the original effect). We explore how the final LayerNorm scaling factor can contribute to self-repair, and additionally discuss the implications of these results for interpretability practitioners.

## 1 INTRODUCTION

Interpretability efforts aim to assign human-understandable behaviors to large language model components. One common technique for doing so is via model ablations, where outputs of individual model components are replaced with ones from other distributions. Ideally, if an attention head is critical for a task, ablating it would substantially worsen model performance.

However, recent work (McGrath et al., 2023; Wang et al., 2023) has uncovered preliminary evidence of self-repair, a phenomena in large language models where components downstream of ablations compensate for them. As such, the ablation of individual components doesn't always lead to easily predictable changes in model performance; instead, the removal of the components behavior can be compensated for by downstream components in a way which masks the loss of the original component.

This is a challenge for interpretability efforts which rely on ablation-based metrics to define the importance of model components. In particular, self-repair mechanisms minimize the impact of ablating components deemed critical by other metrics.

Past literature has looked at self-repair in incomplete settings; we strengthen this prior work by investigating self-repair across the whole pretraining distribution, by focusing on individual attention heads (a smaller change, and thus more surprising to see repaired), and by investigating the mechanisms behind self-repair on the whole distribution. In this work, we find that direct effect self-repair is an imperfect, noisy process which occurs across the full pretraining distribution, even while ablating individual heads (rather than full layers). We next explore one mechanism, changes in LayerNorm normalization factor, which fuels self-repair[1]. We end with some discussion on the implications of these results for interpretability research. All of our code for the experiments used in this paper is provided at `https://github.com/starship006/backup_research`.

## 2 SELF-REPAIR ON THE FULL DISTRIBUTION EXISTS, BUT IS INCOMPLETE AND NOISY

In this section we confirm that, across models, individual attention heads are 'self-repaired' on the general pretraining distribution.

[*]University of Texas at Austin

[1]We also discovered that sparse MLP Erasure induces self-repair. We explore this further in Appendix B.

### 2.1 DEFINING SELF-REPAIR

We first detail how we define direct effect and self-repair. Our methodology aligns with previous research into self-repair (Wang et al., 2023; McGrath et al., 2023), although not mirroring it identically. Let $(x_1, x_2, ..., x_{n-1}, x_n)$ be a sequence of $n$ tokens we pass into a language model. The language model maps each contiguous prefix of input tokens $(x_1, x_2, ..., x_{k-1}, x_k)$ to a final activation $r_k$, ultimately producing $n$ total final activations $(r_1, r_2, ..., r_{n-1}, r_n)$ for each prefix.

We follow Elhage et al. (2021) in calling these the final residual streams. Each final residual stream $r_k$ can be decomposed into the sum of the output of each layer, plus the original embeddings (Elhage et al., 2021). We refer to anything which adds its output into the residual stream as a model component - components can include individual attention heads, neurons, or the embeddings. Formally, for a decomposition into components $C^i$, define $C^i_k$ as the output on position $k$, $r_k = \sum_i C^i_k$.

For a given final residual stream activation $r_k$, the model outputs a distribution of logits $l_k$, obtained from applying a normalization function $LN$ (such as LayerNorm) to the residual stream corresponding to $k$-th position, $res \in R^{d_{model}}$ (where $d_{model}$ is the dimensionality of the model's hidden states), and multiplying it by the model's unembedding matrix $W_U \in R^{V \times d_{model}}$, such that $l_k = W_U \cdot LN(r_k)$.

$W_U$ is a linear map, and as shown by Elhage et al. (2021), we can take a linear approximation to $LN$ on a given input. The composition of two linear functions is a linear function, i.e. a matrix multiplication plus a bias, so $l_k = Wr_k + b$ for some $W$ and $b$ (which vary with the input). Because $r_k$ is the sum of component outputs, $l_k = W(\sum_i C^i_k) + b = (\sum_i WC^i_k) + b$.

We often just care about the logit of the correct next token $x_{k+1}$, which corresponds to a single element of $l_k$. This is $logit_{clean} = W[x_{k+1}]^T \cdot r_k + b[x_{k+1}]$, where $W[x_{k+1}]$ is the $x_{k+1}$-th row of $W$ and $b[x_{k+1}]$ is the $x_{k+1}$-th element of $b$ (which are a $R^{d_{model}}$ vector and a scalar).

Using the decomposition of $l_k$, we can define $W[x_{k+1}]^T \cdot C^i_k$ as the **direct effect** of component $i$ on position $k$. Intuitively, a component can help a model predict the final answer by either producing intermediate representations that are used by later components, or by directly boosting the correct next token: the direct effect captures the latter effect only.

Notice that just as the residual stream can be mostly decomposed into the sum of outputs of all model components, the logit of the correct next token $logit_{clean}$ can be mostly decomposed into the sum of the direct effects of all model components. Ideally, if a component has a high direct effect, it was likely 'important' for predicting the next token.

We contrast the direct effect with ablation-based metrics. Intuitively, ablation-based metrics are designed to measure the full effect of a component on helping the model produce the final answer, both via producing intermediate representations used by later components, and the direct effect. In particular, throughout this paper, we perform a technique called resample ablation, a technique that replaces the output of the same head on a different set of pretraining tokens (Chan et al., 2022). In this ablation, we run the model again on a different set of input tokens $(x'_1, x'_2, ..., x'_{n-1}, x'_n)$ and store the new output of an attention head $C'_{out}$. Then, when running the model on the original set of tokens $(x_1, x_2, ..., x_{n-1}, x_n)$, we causally intervene during the forward pass to the original output of the same attention head $C_{out}$ with this new version $C'_{out}$, and then continue the forward pass, a technique known as activation patching or causal mediation analysis (Vig et al., 2020).

Let the new logit distribution under this ablation be $l'_k$ and the new logit of the correct next token be $logit_{ablated}$. Notice that we can also compute the ablated direct effect of the attention head. If the output of an attention head has no 'indirect' downstream effects—i.e., no downstream model component depends on the output of the attention head—then the change in the correct logit, $\Delta logit = logit_{ablated} - logit_{clean}$, would equal the change in the direct effect of the attention head, $\Delta DE_{head}$. However, in practice, these are rarely equal, and often $|\Delta DE_{head}| > |\Delta logit|$. This discrepancy is caused by later components changing their direct effects in a way that compensates for the resample ablated component, the phenomena we refer to as self-repair. We measure the 'self-repair' that occurs as[2]:

$$\text{self-repair} = \Delta logit - \Delta DE_{head}$$

[2]Note that our definition allows for self-repair to be negative. If the original direct effect of an ablated attention head was negative, we expect the self-repair to also be negative.

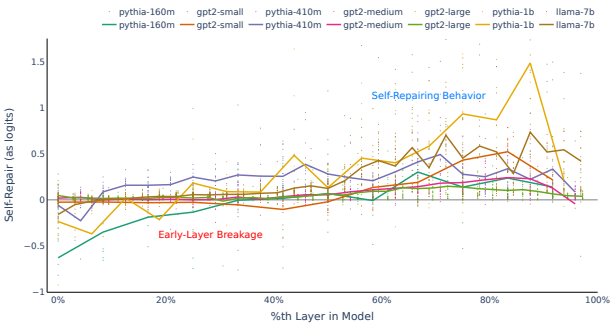


Figure 1: We measure the self-repair of an attention head when resample ablated on the top 2% of tokens according to its direct effect. For each model, we plot both the self-repair of the individual heads and a trend line averaging across the heads in each layer. Self-Repair exists across many later layers in different models, although the amount varies between heads.

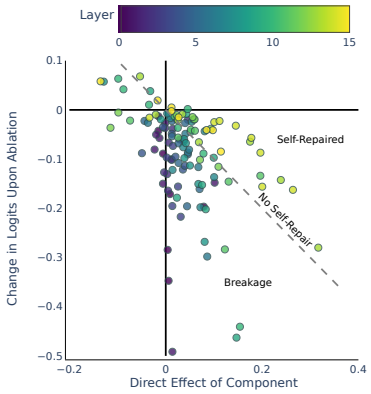


Figure 2: Self-Repair of individual attention heads on Pythia-1b across 1M tokens. For each head in Pythia-1b, we plot its direct effect and the change in logits when reample ablating it. The heads between the included $y = -x$ line and the x-axis are self-repaired.

When we resample ablate a head, the new activation comes from a prompt drawn from the pretraining distribution, and so is likely to be unrelated. The new direct effect when resample ablating an arbitrary head is near zero so $\Delta\text{DE}_{\text{head}} \approx -DE_{\text{head}}$ when resample ablating most attention heads. As such, for convenience we often plot the original direct effect rather than change in direct effect. For example, if an attention head with a direct effect of $0.5$ is ablated and the logit difference is only $-0.2$ logits, we can approximate the self-repair as $-0.2 - (-0.5) = 0.3$.

### 2.2 Self-Repair Exists, but Imperfectly

We measure the self-repair of individual attention heads in Pythia-1b (Biderman et al., 2023) on 1 million tokens of The Pile (Gao et al., 2020), the dataset used to train Pythia-1b [3] . For a given token and forward pass, we measure a head's direct effect $DE_{head}$, as well as the outputted logit of the correct next token. We then resample ablate the attention head (Chan et al., 2022), measure the new logit of the same correct next token, and calculate the logit difference $\Delta$logit. Averaging these values across the whole dataset, we plot each head's direct effect and logit difference in Figure 2 (see Appendix J for the equivalent graphs of other models).

Since $\Delta\text{DE}_{\text{head}} \approx -\text{DE}_{\text{head}}$, heads which aren't self-repaired have $\Delta\text{logit} \approx -\text{DE}_{\text{head}}$, as the change directly feeds into the final logits without any downstream compensation. Heads that *are* self-repaired have smaller logit differences, such that $-1 < \frac{\Delta\text{logit}}{\text{DE}_{\text{head}}} < 0$. In Figure 2, this corresponds to the heads which fall between the $y = -x$ line and the x-axis. It's clear that there are many such heads, even for heads in the last layer (which don't have attention or MLP layers downstream of it[4]).

**The existence of self-repair across the pretraining distribution holds robustly across model sizes and families.** When resample ablating different attention heads in various models, we measure their logit difference $\Delta logit$, change in direct effects $\Delta\text{DE}_{\text{head}}$ , and thus the self-repair experienced.

It has been postulated that each language model component tends to be useful on a sparse set of tokens (Bricken et al., 2023), and we corroborate this here by observing a sparsity of significant direct effect. Accordingly, for each head, we measure the average self-repair each attention head experiences across the top 2% of tokens filtered by each head's direct effect, to ensure we are seeing self-repair of the head's role in the model, rather than noise.

[3]For this experiment and others, unless stated otherwise, we use 1 Million tokens for experiments with the Pythia suite of models and GPT2-Small and GPT2-Medium, but smaller amounts for the other models.

[4]Pythia-1B is a parallel attention model.

We plot the average self-repair experienced by each attention head in Figure 1, along with the mean for all the heads in each layer. It is clear that many heads in later layers of models experience self-repair.

**Self-Repair is Imperfect:** It's important to note that the heads that *are* self-repaired are not perfectly repaired across the entire distribution. Instead, ablating these heads leads to a small, but noticeable, logit difference. This has important implications for self-repair (see Section 4).

### 2.3 Self-Repair is Noisy

Self-Repair is a noisy phenomena on multiple levels. Clean hypotheses such as "the direct effects of all late layer heads are self-repaired by 70%" are immediately falsified (from Figure 1 and 2). We observed several other phenomena once we moved beyond just averaging over many prompts that suggest that self-repair is a noisy, difficult-to-study phenomena that is unlikely to have a single clear mechanistic explanation. These include:

**Self-Repair varies at the level of tokens:** it's not the case that the self-repair systematically exists similarly across prompts/predictions. Averaging self-repair across the full distribution hides a lot of important detail. Indeed, it is instead extremely noisy.

As a case example, in Figure 3 we've plotted the direct effects and logit differences of L22H11 (the 11th head in Layer 22) and L20H6 in Pythia-410m on individual tokens when resample ablated. It is clear that there is immense amounts of noise in the self-repair between individual tokens across the same head, despite being able to observe that these two heads appear to be self-repaired on average.

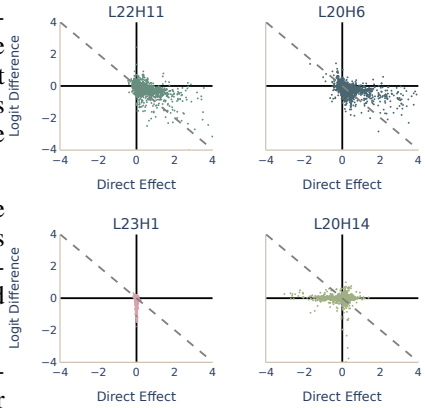


Figure 3: We've selected four heads in Pythia-410m, and plotted the direct effect and logit difference when ablating the head across 5000 individual tokens in The Pile. Within a single head, these values can vary highly. The tokens between the included $y = -x$ line and the x-axis are self-repaired.

**Many heads in the model don't have clear correlations between the direct effect of the head and the change in logits upon ablation.** See L23H1 and L20H14 in Figure 3 for an example of this. For these heads, there is no clear correlation observed between the direct effect of the head and logit difference when resample ablating it. This lack of a clear correlation was first introduced in McGrath et al. 2023 to be a phenomena when ablating layers. We hypothesized that ablating entire layers may have been too drastic of a change (and that smaller scale ablations may be cleaner): however, even at the level of individual attention heads, this result holds. This is unsurprising for heads without direct effects, but surprising for heads which have significant direct effects yet no strong correlation, such as L20H14: it suggests that the direct effect may be nearly fully self-repaired in some heads.

## 3 Nontrivial Self-Repair Due to LayerNorm

What factors may explain self-repair? It turns out that changes in normalizing factors can induce self-repair. If we ablate a head and freeze the output of all downstream components *except* for the LayerNorm normalization factor, this recovers a significant logit difference relative to freezing all components *including* the normalization factor (the direct effect). This is particularly notable, because LayerNorm is often treated as a technicality that can be approximated as linear (or ignored) in mechanistic interpretability work (Elhage et al., 2021)[5].

LayerNorm causes self-repair by acting on the existing logits during an ablation (which we argue for mathematically in Appendix G). The LayerNorm scaling factor scales the existing logits on a forward pass, and if the factor decreases, this can cause an increase in the existing logits as a result. This holds empirically: we take L11H2 in Pythia-160m, which demonstrates this phenomena well,

[5]Note that this repair mechanism applies in the same way to RMSNorm, a popular LayerNorm alternative, which is used in LLaMA models.

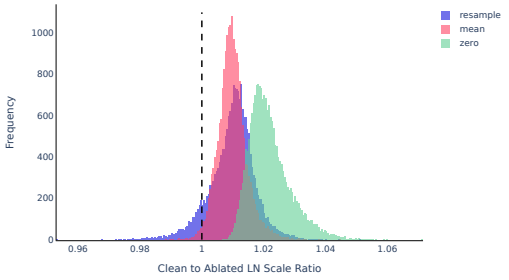


Figure 4: Ratio of clean to ablated LayerNorm scaling factors when resample ablating L11H2 of Pythia-160M, and then filtering for the top 2% of tokens according to direct effect. Ratios greater than 1 indicate that LayerNorm is self-repairing by amplifying the existing logits.

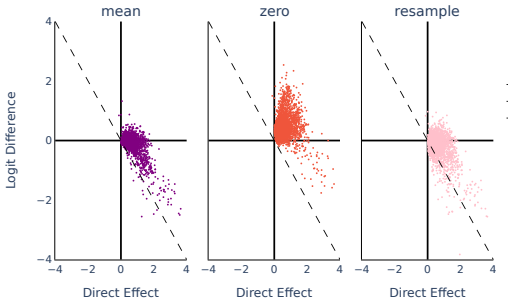


Figure 5: Direct effect vs logit difference of L11H0 in Pythia-160m under different ablations. Notice how zero ablations can induce positive logit differences. This self-repair can only occur due to LayerNorm scale changes.

and ablate it across 1 million tokens on The Pile. On the top 2% of tokens according to its direct effect, we plot the ratio between clean and ablated LayerNorm scaling in Figure 4. We've also included zero and mean ablating, ablations where you replace the output of the head with the zero vector or the average output of the head across a batch, respectively.

The ratio between the clean and ablated LayerNorm scaling factors is almost always greater than one (90.66% of tokens for resample, 98.75% for mean, 100% for zero ablation). This indicates that the existing logits in the ablated run are scaled by less than the logits on the clean run (Appendix G).

This highlights how various forms of common ablations (zero, mean, and resample) can sometimes influence the scaling factors quite significantly. In particular, zero ablating may strongly change the norm of the residual stream. This has practical consequences for the self-repair of attention heads: Figure 5 highlights the self-repair which occurs ablating L11H10 in Pythia-160m with the different ablations. Recall that L11H10 is in the last attention layer of Pythia-160m, and that Pythia-160m is a parallel-attention model: as such, the only component responsible for any self-repair is LayerNorm. Zero-ablating can *increase the logit difference*, even when ablating the head while it has a positive direct effect. Note that it's not always the case that zero ablating has such an extreme effect, see Appendix H. On further efforts to quantitatively measure the extent to which LayerNorm self-repairs, we calculate that LayerNorm can explain around 30% of the self-repair (Appendix C).

## 4 Implications of Imperfect Repair for Interpretability Efforts

The major problem with self-repair is that it makes ablations an unreliable tool for interpretability. In practice, this is most concerning when considering circuit analysis, where we ablate individual model components to isolate the sparse subgraph of the model relevant to our task, as performed in (Wang et al., 2023; Lieberum et al., 2023; Conmy et al., 2023).

Fortunately, circuit analysis only requires identifying whether a component is important or unimportant (i.e. whether it belongs in the sparse subgraph), not the precise effect of ablating it. The fact that self-repair is imperfect (Section 2.2) helps reduce some of the concerns for circuit discovery efforts because the 'importance' of various components is extremely heavy tailed: even a significant fractional decrease in the estimated effect of a node won't change which nodes are important.

However, this doesn't fully alleviate all concerns. In certain situations self-repair can be lossless or overcompensate. This may happen on certain narrow distributions, or may be induced depending on what tools you use. And if the degree of self-repair differs significantly between components, borderline components may be incorrectly included or excluded.

An additional implication of LayerNorm self-repair is that interpretability practitioners should be careful when interpreting the consequences of taking models off-distribution. The fact that self-repair occurs across models tells us something about the internal mechanisms of these models, but it may be a byproduct of some other mechanism in the model, rather than that they are intentionally self-repairing. We discuss this further in Appendix D.

## REFERENCES

David Bau, Jun-Yan Zhu, Hendrik Strobelt, Agata Lapedriza, Bolei Zhou, and Antonio Torralba. Understanding the role of individual units in a deep neural network. *Proceedings of the National Academy of Sciences*, 117(48):30071–30078, September 2020. ISSN 1091-6490. doi: 10.1073/pnas.1907375117. URL http://dx.doi.org/10.1073/pnas.1907375117.

Nora Belrose, Zach Furman, Logan Smith, Danny Halawi, Igor Ostrovsky, Lev McKinney, Stella Biderman, and Jacob Steinhardt. Eliciting latent predictions from transformers with the tuned lens, 2023.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, Usvsn Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar Van Der Wal. Pythia: A suite for analyzing large language models across training and scaling. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 2397–2430. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/biderman23a.html.

Steven Bills, Nick Cammarata, Dan Mossing, Henk Tillman, Leo Gao, Gabriel Goh, Ilya Sutskever, Jan Leike, Jeff Wu, and William Saunders. Language models can explain neurons in language models. https://openaipublic.blob.core.windows.net/neuron-explainer/paper/index.html, 2023.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E Burke, Tristan Hume, Shan Carter, Tom Henighan, and Christopher Olah. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. https://transformer-circuits.pub/2023/monosemantic-features/index.html.

Lawrence Chan, Adrià Garriga-Alonso, Nicholas Goldwosky-Dill, Ryan Greenblatt, Jenny Nitishinskaya, Ansh Radhakrishnan, Buck Shlegeris, and Nate Thomas. Causal scrubbing, a method for rigorously testing interpretability hypotheses. *AI Alignment Forum*, 2022. https://www.alignmentforum.org/posts/JvZhhzycHu2Yd57RN/causal-scrubbing-a-method-for-rigorously-testing.

Arthur Conmy, Augustine N. Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. Towards automated circuit discovery for mechanistic interpretability. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=89ia77nZ8u.

Guy Dar, Mor Geva, Ankit Gupta, and Jonathan Berant. Analyzing transformers in embedding space, 2023.

Mostafa Dehghani, Stephan Gouws, Oriol Vinyals, Jakob Uszkoreit, and Łukasz Kaiser. Universal transformers, 2019.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. https://transformer-circuits.pub/2021/framework/index.html.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. The Pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020.

Nicholas Goldowsky-Dill, Chris MacLeod, Lucas Sato, and Aryaman Arora. Localizing model behavior with path patching, 2023.

Rhys Gould, Euan Ong, George Ogden, and Arthur Conmy. Successor heads: Recurring, interpretable attention heads in the wild, 2023.

Klaus Greff, Rupesh K. Srivastava, and Jürgen Schmidhuber. Highway and residual networks learn unrolled iterative estimation, 2017.

Wes Gurnee, Neel Nanda, Matthew Pauly, Katherine Harvey, Dmitrii Troitskii, and Dimitris Bertsimas. Finding neurons in a haystack: Case studies with sparse probing, 2023.

Wes Gurnee, Theo Horsley, Zifan Carl Guo, Tara Rezaei Kheirkhah, Qinyi Sun, Will Hathaway, Neel Nanda, and Dimitris Bertsimas. Universal neurons in gpt2 language models, 2024.

Stefan Heimersheim and Alex Turner. Residual stream norms grow exponentially over the forward pass, 2023. URL https://www.alignmentforum.org/posts/8mizBCm3dyc432nK8/residual-stream-norms-grow-exponentially-over-the-forward.

Stanisław Jastrzebski, Devansh Arpit, Nicolas Ballas, Vikas Verma, Tong Che, and Yoshua Bengio. Residual connections encourage iterative inference, 2018.

Matthew L. Leavitt and Ari Morcos. Towards falsifiable interpretability research, 2020.

Tom Lieberum, Matthew Rahtz, János Kramár, Neel Nanda, Geoffrey Irving, Rohin Shah, and Vladimir Mikulik. Does circuit analysis interpretability scale? evidence from multiple choice capabilities in chinchilla, 2023.

Callum McDougall, Arthur Conmy, Cody Rushing, Thomas McGrath, and Neel Nanda. Copy suppression: Comprehensively understanding an attention head, 2023.

Thomas McGrath, Matthew Rahtz, Janos Kramar, Vladimir Mikulik, and Shane Legg. The hydra effect: Emergent self-repair in language model computations, 2023.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt, 2023.

Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. Progress measures for grokking via mechanistic interpretability. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=9XFSbDPmdW.

nostalgebraist. interpreting gpt: the logit lens, 2020. URL https://www.lesswrong.com/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens.

Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. Zoom in: An introduction to circuits. *Distill*, 2020. doi: 10.23915/distill.00024.001. https://distill.pub/2020/circuits/zoom-in.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. In-context learning and induction heads. *Transformer Circuits Thread*, 2022. https://transformer-circuits.pub/2022/in-context-learning-and-induction-heads/index.html.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019. URL https://api.semanticscholar.org/CorpusID:160025533.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Simas Sakenis, Jason Huang, Yaron Singer, and Stuart Shieber. Causal mediation analysis for interpreting neural nlp: The case of gender bias, 2020.

Elena Voita, Javier Ferrando, and Christoforos Nalmpantis. Neurons in large language models: Dead, n-gram, positional, 2023.

Kevin Ro Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: a circuit for indirect object identification in GPT-2 small. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=NpsVSN6o4ul.

# A Related Work

The **self-repair phenomena** was first identified in the IOI distribution Wang et al. (2023). Self-Repair was initially explored as the Hydra Effect McGrath et al. (2023), and a specific instance of it was explained by Copy Suppression McDougall et al. (2023).

Understanding self-repair fits in the broader work of **Mechanistic Interpretability**, which aims to reverse engineer neural networks. Related work includes Olah et al. 2020 on vision models and Meng et al. 2023 on Transformer models. Previous work has attempted to understand the behavior of individual neurons (Bau et al., 2020; Gurnee et al., 2023) or attention heads (Gould et al., 2023). In particular, suppression neurons were previously found in Voita et al. 2023 and Gurnee et al. 2024. One important aspect of Mechanistic Interpretability is automating circuit discovery (Conmy et al., 2023; Bills et al., 2023).

Recent research has emphasized the importance of using casual mechanisms to measure component importance (Chan et al., 2022). Ablations have been a common technique for this (Leavitt & Morcos, 2020), and have been used to validate hypothesis in Olsson et al. 2022; Nanda et al. 2023.

Ideas related to the **Model Iterativity Hypothesis** were introduced in Greff et al. 2017; Jastrzebski et al. 2018. The Universal Transformer Dehghani et al. (2019), Logit Lens tool nostalgebraist (2020), and Tuned Lens tool Belrose et al. (2023) were built on ideas similar to this.

# B Sparse Neuron Anti-Erasure helps Self-Repair

McGrath et al. 2023 identified MLP Erasure, a behavior in MLP layers where important directions outputted by earlier components are written against by later MLP components. This exists in the context of direct effects: the direct effect of components will occasionally be negated by downstream components, 'erasing' parts of the existing direct effects.

However, since the erasure behavior is dependent on the upstream components, this creates a form of "Anti-Erasure" self-repair where the removal of the upstream component's direct effect also induces the removal of the downstream Erasure. A similar motif was discovered in Copy Suppression attention heads (McDougall et al., 2023), where the ablation of specific attention heads caused the Copy Suppression heads to perform less Erasure. We build upon this initial work and highlight that MLP Erasure is fueled by a sparse set of Erasure neurons, which change between prompts.

## B.1 Erasure Occurs in Neurons

McGrath et al. 2023 introduced MLP Erasure as a behavior performed by MLP layers that caused self-repair. However, we hypothesized that MLP layers were too broad of a unit of analysis: we narrow our focus to the level of neurons. We find that self-repair occurs due to Anti-Erasure in Neurons.

To provide an example, we take L10H11 in Pythia-160m, an attention head right before the final MLP layer, and resample ablate the head across The Pile. Then, we isolate the top 2% of instances in which the last MLP layer displays the most self-repair. For each token, we isolate the individual neuron which self-repairs the most in the layer (for that instance), and plot its clean and ablated direct effect, colored by the direct effect of the entire last MLP layer, in Figure 6.

We observe that on across many tokens, the top self-repairing neuron has a negative clean direct effect and a less negative ablated direct effect upon ablation of L10H11. This indicates that it was originally performing erasure, but performed less erasure upon the ablation of L10H11. This is in contrast to the direct effects of the entire last layer, which are frequently extremely positive (as indicated by the darker blue color of the majority of the points in Figure 6).

## B.2 Sparse, Different Sets of Neurons Self-Repair

In the instances where significant MLP self-repair is occurring, it seems to be dominated by a few significant neurons.

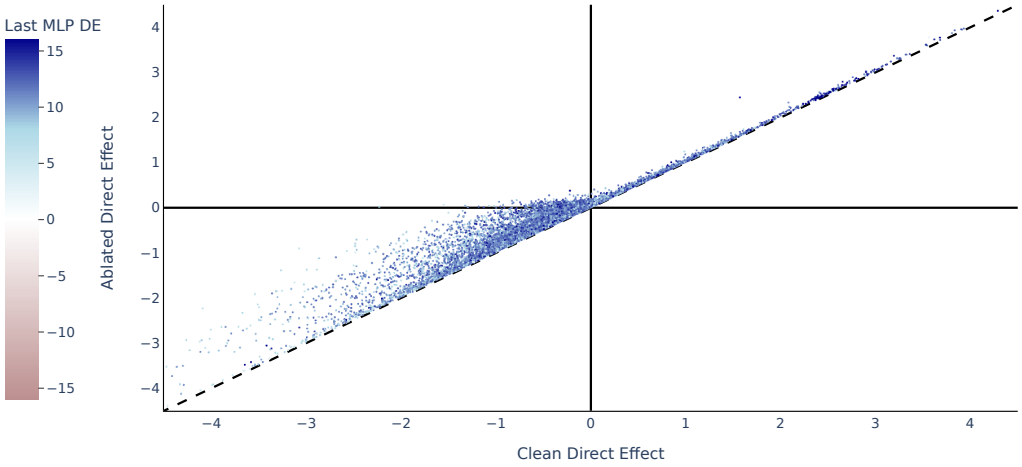


Figure 6: For selected tokens on The Pile, we plot the clean and ablated direct effect of the top self-repairing neuron in the last MLP layer when resample ablating L10H11 in Pythia-160m. Each token is colored with the direct effect of the entire MLP layer on the clean run.

For four different models, we resample ablate a head in a later layer which has a large average direct effect across tokens from The Pile, and see how much the top self-repairing neurons self-repair. For each head, we isolate the 2% of tokens on The Pile for which the head has highest direct effect, and measure the self-repair that occurs in the very last MLP layer of the model when resample ablating the head. Then, for each token, we measured how much the top $N$th self-repairing neuron in the last layer self-repaired as a fraction of the direct effect of the head. We plot the percentage of instances for which the *neuron* explained 50, 10, 5, 2, and 1% of the direct effect of the head in Figure 7.

It's clear that a few neurons can explain disproportionate amounts of the total self-repair: recall that Llama-7B (Touvron et al., 2023), GPT2-Small (Radford et al., 2019), Pythia-160m, and Pythia-410m have 11008, 4096, 3072, and 3072 neurons per layer, respectively. As such, across all of these examples, only a sparse set of neurons have significant amounts of self-repair.

It's important to note that many neurons have marginally changed direct effects as a result of the ablation. The self-repair of the top neurons may be significant on a specific instance, but this doesn't mean that they are the *only* self-repairing neurons: there may be many other neurons with changed direct effects, both positively and negatively.

Are there neurons that perform Erasure and self-repair consistently? Anecdotally we observed that the same neurons can often self-repair across the same prompt, but across different prompts, they may differ. We filter for the top 2% of tokens where L10H11 in Pythia-160M is self-repaired the most and collect the top 10 repairing neurons in the last layer, per prompt. The most any single neuron is in the top-10 self-repairing neurons is 16% of the tokens.

This suggests that there are likely different neurons responsible for different forms of erasure. In particular, this is potentially related to "Suppression Neurons", discovered to decrease probabilities of related tokens Gurnee et al. (2024). This would explain why neurons may similarly self-repair across the same prompt, but not on the entire distribution.

## C Quantifying LayerNorm Self-Repair

To attempt to quantitatively measure the extent to which LayerNorm self-repairs, we breakdown the self-repair experienced by each head into a LayerNorm, MLP, and attention head components as follows:

$$\begin{aligned} \text{self repair} &= \Delta\text{DE}_{\text{head}} - \Delta\text{logit} \\ &= \sum_{h \in H} \Delta\text{DE}_h + \sum_{m \in M} \Delta\text{DE}_m + \Delta\text{DE}_{\text{LayerNorm}} \end{aligned} \tag{1}$$

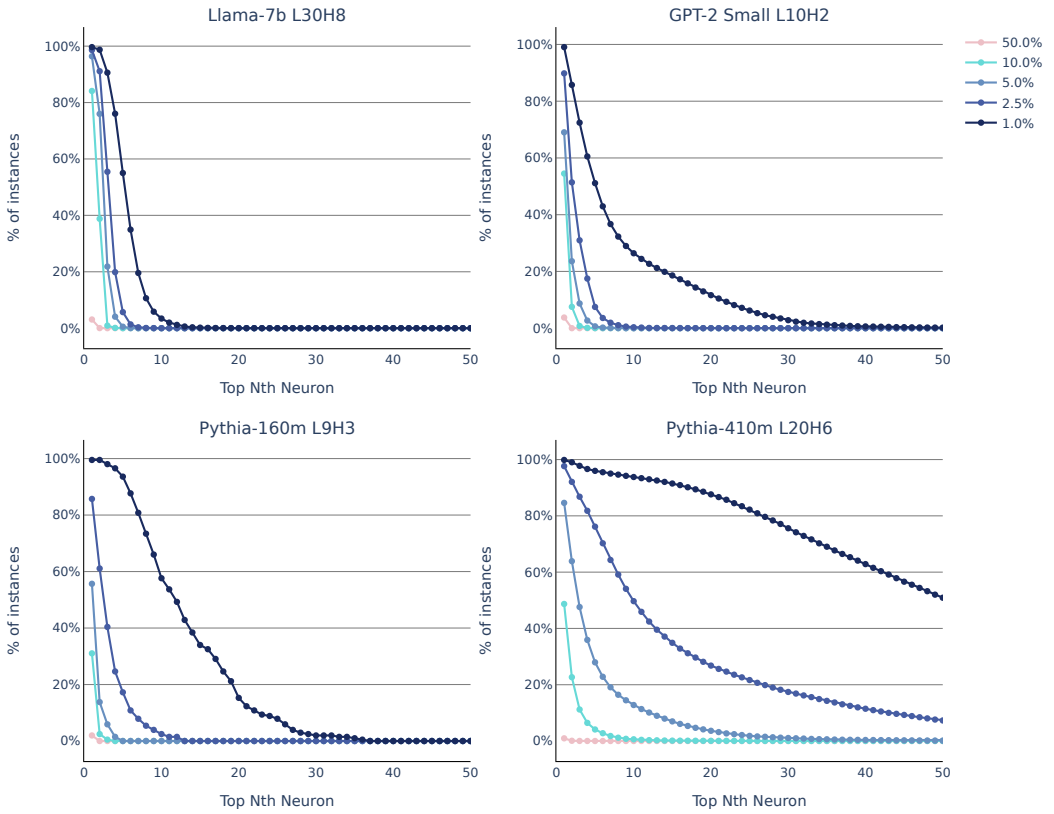


Figure 7: For four models and attention heads, we plot the frequency at which the top $N$th self-repairing neuron repaired X% of the direct effect (on the top 2% of the tokens by direct effect).

where we are measuring the sum of the:

1. Changes in direct effect $\Delta\text{DE}_h$ for each head $h$ in the set $H$ of all heads downstream of the ablated head.
2. Changes in the direct effect $\Delta\text{DE}_m$ for each MLP layer $m$ in the set $M$ of all MLP layers downstream of the ablated head.

and using the difference between the calculated self-repair and this sum to determine how much of the remaining self-repair is due to changes in the final LayerNorm scale $\Delta\text{DE}_{\text{LayerNorm}}$. This ends up accounting for the effect of the LayerNorm scale on both the existing logits and the changes in all direct effects.

Across models, we calculate these values from Equation 1 for each attention head on the top 2% of tokens in The Pile according to their direct effect. The self-repair due to LayerNorm is plotted in Figure 8.

As self-repair is noisy (Section 2.3), creating a summary statistic to capture how much self-repair explains the direct effect of a head is extremely difficult. Measuring self-repair as a fraction of the direct effect often creates extreme values which are uninterpretable and not useful.

But, when measuring the self-repair due to LayerNorm on a token as a fraction of the clean direct effect of the head on that token, *and capping the percentage between 0 and 100%*, we learn that LayerNorm can explain around 30% of the direct effect of a head on average across many layers. This is best treated more as a directional metric highlighting the existence of this motif, rather than a concrete, precise one; we illuminate the difficulty in precisely measuring self-repair as fractions of direct effects, and our approaches towards dealing with this, in more detail in Appendix F.

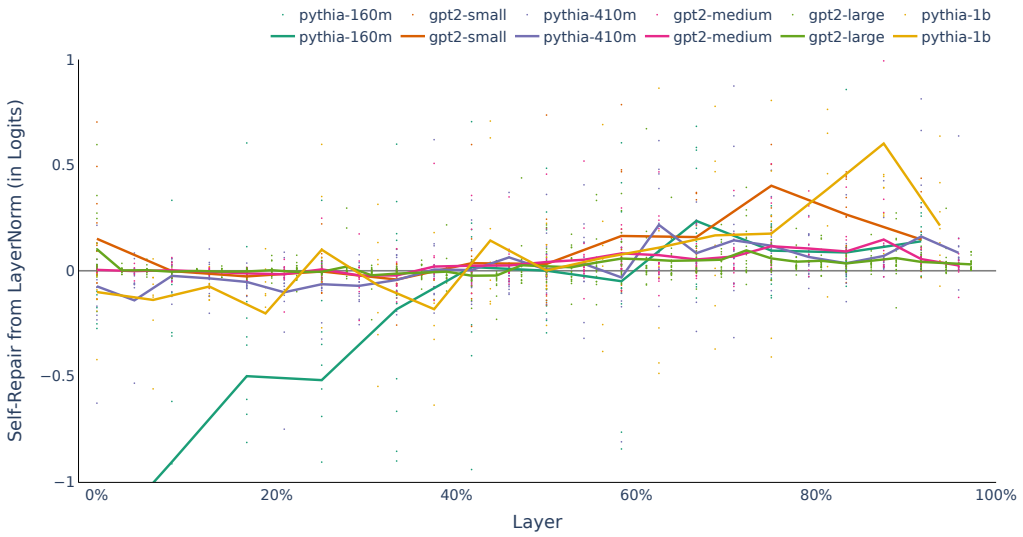


Figure 8: Across the top 2% of tokens on The Pile, we measure the LayerNorm self-repair when resample ablating individual heads. We've plotted mean lines averaging the values across each model's layer. LayerNorm has a significant effect on later layers of the model.

## D Taking Models Off-Distribution can Cause Unexpected Consequences that are Easy to Misinterpret

One reason to expect self-repair to occur is that taking the model off-distribution can lead to very surprising consequences. Ablations fundamentally take the model off-distribution, in the sense that the set of internal activations achieved from ablations may be impossible to achieve on *any* input. Models were not trained to respond coherently to these kinds of internal interventions, and so may behave in erratic and hard-to-predict ways which are difficult to reason about.

When we first found self-repair, we assumed that models were intentionally self-repairing. However, the LayerNorm based self-repair seems to be a side effect of unrelated mathematical properties of the model, interacting with how our ablations change the norm of the residual stream. It's possible that many observed instances of 'self-repair' are just uninteresting consequences of throwing the model off-distribution, for which this will be clear after discovering a few more insights like the above.

Importantly, the fact that self-repair occurs across models tells us something about the internal mechanisms of these models, but it may be a byproduct of some other mechanism in the model, rather than that they are intentionally self-repairing.

This is a problem fundamental to ablations, and is likely difficult to circumvent. A suggested course of action is to try to control how far from the standard distribution one's causal interventions take the model Chan et al. (2022). Two ways this can be done is by avoiding zero ablation (which most significantly adjusts the residual stream norm, as shown in Section 3) or by freezing LayerNorm while ablating. Additionally, one could use path patching (Goldowsky-Dill et al., 2023) instead of patching full components.

Our results do not demonstrate that all of self-repair is byproducts of other mechanisms: LayerNorm self-repair only explains up to around 30% of the self-repair on average. But these results are a word of caution to be careful when taking models off distribution.

## E Model Iterativity Hypothesis

There's a wealth of evidence (nostalgebraist, 2020; Dar et al., 2023) that models gradually build up their final logits in unembedding space: the correct token isn't predicted by a singular model

component, but rather built up over time by the outputs of multiple components. An implication of this is that models can still be relatively accurate, even when removing the final layer.

One perspective from which self-repair becomes less surprising is what we call the Model Iterativity Hypothesis: rather than layers being part of a top-down process, assisting all future layers in complex circuits, many language models components are more of a bottom-up process, where each layer treats the input as a guess for the final output, and tries to reduce the error between this guess and the true next token.

From this perspective, self-repair is unsurprising. Imagine that some task, such as Name Moving, must be performed. Some earlier component in the model will write the signal into the residual stream that a task $T$ needs to completed. If the Model Iterativity Hypothesis holds, then rather than there being some dedicated head to do the task, there are many such capable heads: the earliest one that reads in the signal will perform the task $T$, and likely write the signal that the task is complete. Importantly, if the head that performed task $T$ was ablated, then the need to complete task $T$ would still exist (as well as the associated information of it): as such, it is possible for a downstream head to observe and complete it instead.

This is consistent with the evidence presented in McDougall et al. 2023 on how specific heads can influence downstream heads to not perform a task. We present additional evidence for and against the Model Iterativity Hypothesis in Appendix E.1, and highlight one important line below. One line of evidence that supports this hypothesis is the identification of attention heads across models which we dub 'self-reinforcing' and 'self-repressing'.

For a specific attention head, we take its output on a forward pass, and re-run the forward pass while adding the output of the head *back into the residual stream which feeds into the head*. We measure the original and new direct effect as a result of this intervention.

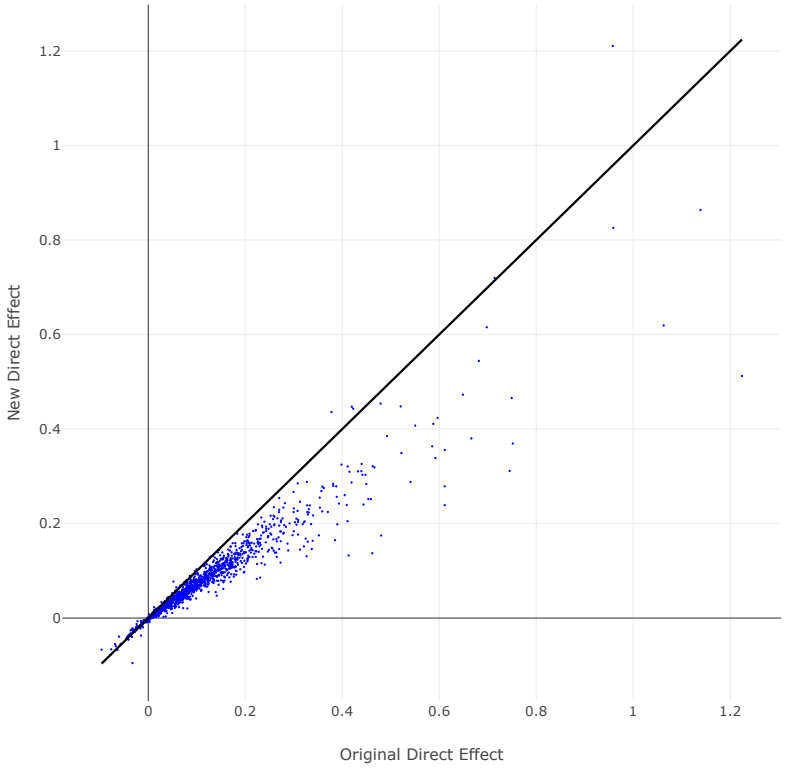


Figure 9: A self-repressing head in GPT2-Medium is L21H1. We plot L21H1's original and new Direct Effects on different tokens after adding its output, scaled by three, back into the previous residual stream. L21H1 has a decreased direct effect as a result.

We observe 'self-reinforcing' and 'self-repressing' heads, in which the direct effect of the head increases or decreases proportional to how much of the original output of the head is added. Figure

9 highlights one head in GPT2-Medium which 'self-represses'. The repression is proportional to the amount the output is scaled (Appendix I).

Not all heads are self-reinforcing and self-repressing. Some appear to be combinations of the two. However, the presence of self-repressing heads suggests that certain attention heads may be outputting a signal that specifies that the task they performed is not needed anymore.

### E.1 More Evidence For and Against Model Iterativity Hypothesis

An additional reason to think that the Model Iterativity Hypothesis is true is that different attention heads in different layers can perform similar tasks.

The Indirect Object Identification task (Wang et al., 2023) highlights one of the earliest instances of observed self-repair. In it, some attention heads are classified as "Name Mover Heads" and other heads as "Backup Name Mover Heads". Originally, it was believed that these were meaningfully different components: however, when running the model across a more general distribution, we can observe instances of the Backup Name Mover Heads performing name moving as well.

The "Name Mover Heads" exist in earlier layers than the "Backup Name Mover Heads". The fact these "Backup" heads can perform moving behavior suggests that they have the capability to perform Name Moving, but don't do so in the Indirect Object Identification task. As a result, ablating the Name Mover Heads would additionally preserve the associated signal to perform Name Moving, which the Backup Heads read in and perform as a result.

Additionally, when moving the residual stream read by the "Name Movers" into the residual stream read by the "Backup Name Movers', the backup heads begin performing name moving as well (note that this is equivalent to zero ablating all the attention heads and MLP layers in between them). This suggests that there is a shared stimulus to perform Name Moving which both types of heads respond to, but which the Name Movers often respond to and ablate.

However, there is also a wealth of evidence against this hypothesis. Anecdotally, previous token and induction heads don't seem to be self-repaired. Perhaps this may be because models expect for these kind of heads to perform this 'fundamental' task. Further, prior literature has emphasized the presence of complex circuits within the model, which don't easily exist across layers, the S-inhibition heads in the Indirect Object Idenficiation task (Wang et al., 2023) being one of them.

This highlights how specific tasks in the head may not be performed 'iteratively': they may be too fundamental such that the model has specific heads dedicated to them, or it may be too complex/costly to perform iteratively by multiple heads. Many heads in the model may not be performing 'iterative' tasks.

## F Troubles with Quantifying Self-Repair Relative To Direct Effects

As we highlighted in Section 2.3, on many occasions the direct effect of a head is sparse. This means that often, measuring self-repair as a fraction of the total direct effect can lead to extreme values which, when averaged with others, skew the data immensely.

As an example, Figure 10 replicates the experiment in Section 3 where we quantify the amount of self-repair LayerNorm explains, but instead purely as a fraction of the total direct effect per head. Here we average across the entire distribution and do not clip the percentages for each token between 0 and 1. It's clear the values are extremely obtuse.

Thus, we attempt to create a cleaner representation of this result: our general two techniques for trying to capture more meaningful summary statistics were to 1) filter for the top instances of direct effect 2) clip the percentages of each token. These are nontrivial changes, and they have important impacts on the final figures we achieve.

A full figure with these changes in effect is shown in Figure 11, which is where we get our observation that "30% of direct effect is self-repaired by LayerNorm": most of the models, on many of the layers, have an average self-repair due to LayerNorm greater than 30% across the heads. The clipping on the values partially means that our figure for "30% of direct effect is self-repaired by


Figure 10: 'Unclipped' Measuring of LayerNorm as a fraction of direct effect

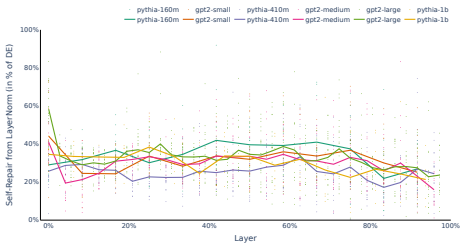


Figure 11: Post-processed Measuring of LayerNorm as a fraction of direct effect

LayerNorm" can potentially be interpreted as "in 30% of cases, the self-repair due to LayerNorm is larger in magnitude than the direct effect of the head", given that the self-repair, on many instances, is larger in magnitude than the actual direct effect of the head.

The noisiness of self-repair also means that even averaging across a layer hides a significant amount of nuance: the various attention heads have vastly different values. As such, we've plotted the individual heads in many of the graphs. However, summary statistics do not capture this nuance well.

## G Mathematical Derivation of LayerNorm Self-Repair

We show the derivation of LayerNorm's self-repair in full. Consider the simplified case where we analyze a parallel attention model such that ablating an attention head in the last layer only has a direct effect on the final residual stream. For an arbitrary model which uses LayerNorm, the model passes in the final residual stream $resid$ into the LayerNorm $LN$, and the logit for the correct token $logit$ is calculated by dotting the result with a vector $L$ which corresponds to the unembedding direction of the logit.

$$logit = \langle LN(resid), L \rangle$$

Assume we resample ablate an attention head in the last layer of attention heads with direct effect $DE_{\text{Head}}$. Consider the direct change in the residual stream, such that this change directly feeds into the new final residual stream $resid'$ with no other intermediate effects. This new residual stream maps onto the new correct token logit $logit' = \langle LN(resid'), L \rangle$. If the change in the output of the head is $\Delta H$, then:

$$resid' = resid + \Delta H$$

Originally, one may predict that the change in logits $\Delta logit = logit' - logit$ is equal to $-DE_{\text{Head}}$, which models the ablation occuring in the absense of self-repair. Self-Repair is the observation that this isn't the case, and that often $|\Delta logit| < |DE_{\text{Head}}|$.

How can we explain this logit difference? As argued in Elhage et al. 2021, we can simplify the LayerNorm operation such that we 'fold in' LayerNorm projections to the weights of linear layers before and after the projection, leaving the nonlinear operation of scaling the residual stream by dividing by a scaling factor proportional to its norm.

The two scaling factors on the clean and ablated run are $S$ and $S'$, which completely describe the LayerNorm functionality. With this, we can model what makes up the difference between the two logits:

$$\Delta logit = logit' - logit = \langle \frac{resid'}{S'}, L \rangle - logit$$

Let's declare the change between $resid' - resid = \Delta H$, which is the difference in the output of head H as a result of ablating the attention head. As such,

$$\Delta logit = (\langle \frac{resid}{S'}, L \rangle + \langle \frac{\Delta H}{S'}, L \rangle) - logit$$

And thus,

$$\Delta logit = (\frac{S}{S'})logit + \langle \frac{\Delta H}{S'}, L \rangle - logit$$

The difference between the two residual streams is only the difference between the outputs between the clean and resample ablated head, $\Delta H = H_{new} - H_{old}$. If you define the new direct effect of the resample ablated head as $DE'_{\text{Head}} = \langle \frac{H_{new}}{S}, L \rangle$, you can rewrite the above as

$$\Delta logit = (\frac{S}{S'} - 1)logit + (\frac{S}{S'})(\langle \frac{H_{new}}{S}, L \rangle - \langle \frac{H_{old}}{S}, L \rangle)$$

And thus,

$$\Delta \text{logit} = \underbrace{(\frac{S}{S'} - 1)\text{logit}}_{\text{LN on } \textit{existing logits}} + \underbrace{\left(\frac{S}{S'}\right)(DE'_{\text{Head}} - DE_{\text{Head}})}_{\text{LN on Expected Change} \Delta DE_{\text{Head}}}$$

## H Residual Stream Norms Change when Ablating

Indeed, the residual stream norm changes most significantly as a result of zero ablations. In Figure 12, we plot the ratio of ablated to clean residual stream norm for different ablations. Even without filtering for different direct effects, you can clearly see how zero ablating decreases the direct effect of a single head.

It's not surprising that a head in the last layer can decrease the residual stream norm by around $5\%$ when ablated. Past work found that residual stream norms grown exponentially (Heimersheim & Turner, 2023), so we would expect heads in later layers to be important for increasing norm.

For some heads, zero ablations induce more extreme changes on the LayerNorm: see additionally L11H3 of GPT-2 Small, plotted in 14. However, this does not always hold. For instance, all the different ablations seem to induce similar amounts of LN scaling ratios in L11H11 of GPT2-Small, plotted in 13.

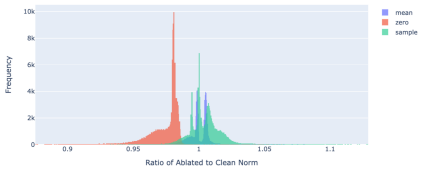


Figure 12: Ratio of ablated to clean residual stream norms when ablating L11H0 of Pythia-160m with various types of ablations, on The Pile.

## I Self-Repressing Heads

The existence of Self-Repressing heads (Section E) helps support the Model Iterativity hypothesis because they robustly respond to their own output. However, one interesting aspect of this is that the amount in which these heads self-repress is proportional to the amount in which scale the heads output back into itself. Figure 15 highlights the self-repressing head L21H1 in GPT2-Medium, but plots the forward passes when adding in the output of the head scaled by 1, 3, and 5 times the existing output.

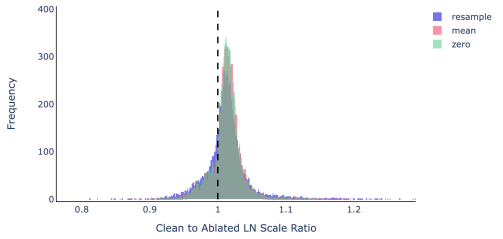


Figure 13: LayerNorm scaling changes from ablating GPT2-Small L11H10. We follow the same experimental procedure outlined in Figure 4.

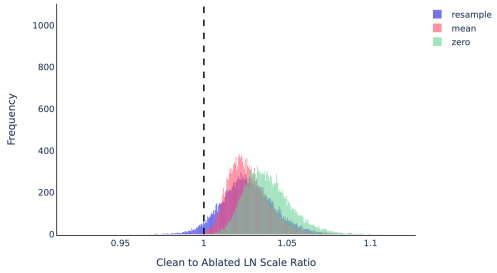


Figure 14: LayerNorm scaling changes from ablating GPT2-Small L11H3. We follow the same experimental procedure outlined in Figure 4.

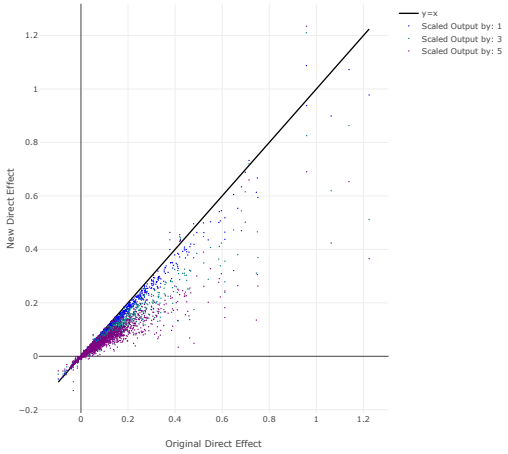


Figure 15: Self-Repressing Head L21H1 in GPT2-Medium being self-repressed by various amounts when amplifying the input into the residual stream by various amounts.

## J SELF-REPAIR GRAPHS ACROSS DIFFERENT MODELS.

We plot the self-repair of individual attention heads for different models, similar to the experimental setup for Figure 2.

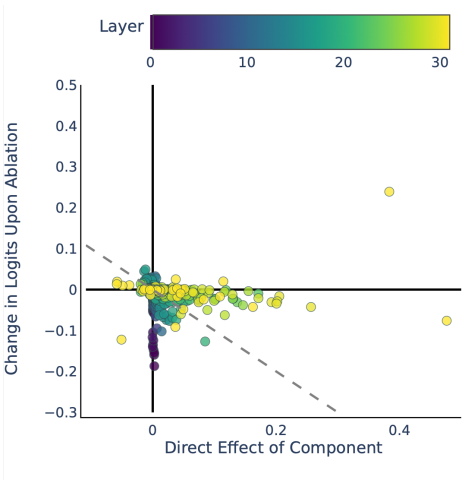


Figure 16: Llama-7b self-repair, per head

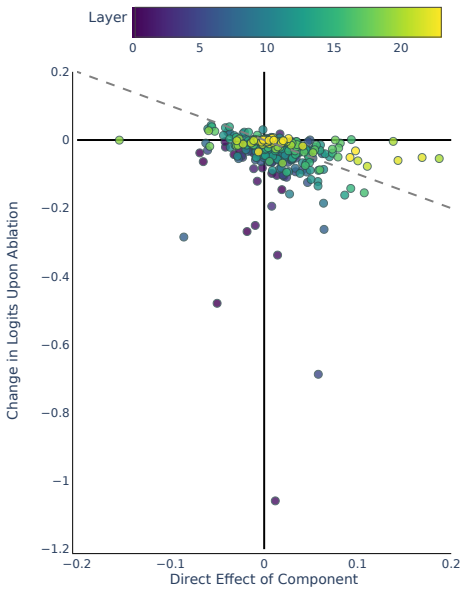


Figure 17: Pythia-410m self-repair, per head

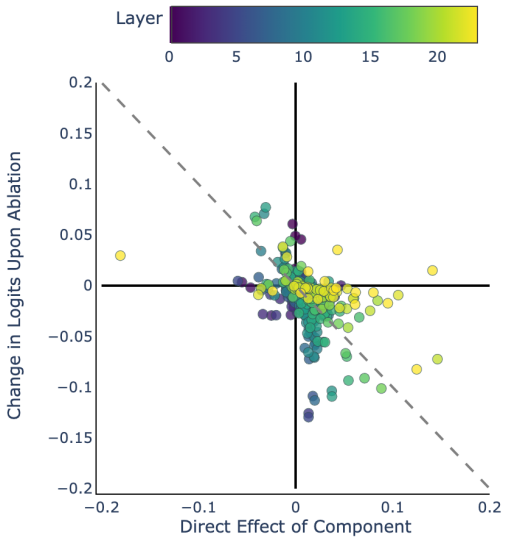


Figure 18: GPT2-Medium self-repair, per head