Missed Causes and Ambiguous Effects: Counterfactuals Pose Challenges for Interpreting Neural Networks

Aaron Mueller [1 2]

Abstract

Interpretability research takes counterfactual theories of causality for granted. Most causal methods rely on counterfactual interventions to inputs or the activations of particular model components, followed by observations of the change in models' output logits or behaviors. While this yields more faithful evidence than correlational methods, counterfactuals nonetheless have key problems that bias our findings in specific and predictable ways. Specifically, (i) counterfactual theories do not effectively capture multiple independently sufficient causes of the same effect, which leads us to miss certain causes entirely; and (ii) counterfactual dependencies in neural networks are generally not transitive, which complicates methods for extracting and interpreting causal graphs from neural networks. We discuss the implications of these challenges for interpretability researchers and propose concrete suggestions for future work.

1. Introduction

Causal interpretability techniques have become popular, in large part due to the increasing popularity of mechanistic interpretability. Such techniques aim to faithfully understand the *causal mechanisms* underlying an observed model behavior. In practice, "causal method" is generally synonymous with "method that employs counterfactual interventions"; assumptions like these take the **counterfactual theory of causality** (Lewis, 1973) for granted, often assuming its benefits but ignoring its pitfalls. This is problematic: counterfactual theories, while intuitive at first glance, leave us prone to counterintuitive fallacies that can systematically bias our results and interpretations if we are not careful.

In this paper, we will argue that there are two tricky problems that arise when employing counterfactual interventions to interpret neural networks: (i) **overdetermination**, where multiple simultaneous and similar causes of the same effect will be systematically missed; and (ii) the **non-transitivity** of counterfactual dependencies, which make it difficult to interpret the relationship between pairs of nodes that are not directly connected. We will start by describing the counterfactual theory of causality (§2) and its role in interpretability research (§2.1). Then, we will describe the resulting problems that have permeated interpretability research (§3, 4). Finally, we will discuss the implications of these problems and propose suggestions for future work (§5).

2. The Counterfactual Theory of Causality

The first counterfactual interpretation of causality was posed by David Hume: "We may define a cause to be an object…where, if the first object had not been, the second had never existed" (Hume, 1748). This was later formalized by Lewis (1973) into what we now recognize as the counterfactual theory. His theory is based on **counterfactual conditionals**, which take the form "If cause C, then event E." These conditionals do not necessarily indicate causation (e.g., "If I were in Berlin, I would be in Germany" is a valid counterfactual conditional, but the inverse does not hold).

Lewis (1973) poses that a **causal dependence** holds iff the following condition holds:

> "An event E *causally depends* on C [iff] (i) if C had occurred, then E would have occurred, and (ii) if C had not occurred, then E would not have occurred."

Lewis (1986) notes that counterfactual dependence is sufficient *but not necessary* for causation; he therefore extends his definition of causal dependence to be whether there is a *causal chain* linking C to E. This approach was later extended from a binary notion of whether the effect happens to a more nuanced notion of causes having influence on *how* or *when* events occur (Lewis, 2000). Other work extends notions of cause and effect to continuous measurable quantities (Pearl, 2000), including direct and indirect effects (Robins & Greenland, 1992; Pearl, 2001); such work generally bases its computations on structural equation modeling

[1]Northeastern University [2]Technion – Israel Institute of Technology. Correspondence to: Aaron Mueller <aa.mueller@northeastern.edu>.

Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

(Goldberger, 1972).

In the causality literature, a causal chain from C to E is referred to as a **mechanism**. The field of **mechanistic interpretability** does not use the term "mechanistic" in this way (or in any standard way; Andreas, 2024), but many of its methods do fit this definition: they aim to locate the causal chain that explains how an input is transformed into a given model behavior.[1]

2.1. Counterfactual Dependencies in Mechanistic Interpretability

When an interpretability study describes its methods as "causal", this typically refers to its use of **counterfactual interventions**. Here, we will use "counterfactual" to refer to any activation that a model component would not naturally take given input x without intervention.[2]

In this paradigm, we conceptualize a neural network model M as a causal graph from inputs to outputs, where each model component (e.g., a neuron or an attention head) is a **mediator** or **node**, and each connection between components in the computation graph is a causal **edge**. This follows terminology and modeling practices in **causal mediation analysis** (Pearl, 2001). Neural networks are amenable to this interpretation because they are *complete* causal graphs that fully explain how any valid input x will be transformed into a corresponding output y.

A typical method for implicating a model component in some behavior is **activation patching** (Vig et al., 2020; Finlayson et al., 2021; Meng et al., 2022). In this method, we first define an input x_{original} and a corresponding correct completion. If we are using a **patching ablation**, we also define a minimally different incorrect completion. For example, in subject–verb agreement, if the input x_{original} is "The **manager**", a minimal pair of completions could be y_{original} = "**is**" (correct) and y_{patch} = "**are**" (incorrect). One then defines a minimally different input x_{patch} where the correct and incorrect answer are flipped; in this example, x_{patch} would be "The **managers**".

Given these variables, one runs x_{patch} through M in a forward pass (denoted $M(x_{\text{patch}})$) and collects the activation a_{patch} of a model component a. Then, one can run $M(x_{\text{original}})$ and collect some output metric that should be sensitive to this component's activation; often, this metric is the logit difference $y = (p_{\text{patch}} - p_{\text{original}})$. Then, one reruns $M(x_{\text{clean}})$, but where a_{original} is replaced with a_{patch}; i.e., we perform a counterfactual intervention to a by setting its value to what it *would have been* in an alternate input. One then measures how much this intervention changes the target metric y.[3]

This change in the output metric is the **indirect effect** (IE; Pearl, 2001; Robins & Greenland, 1992) of a on y. The IE after activation patching is a common metric in causal interpretability (Vig et al., 2020; Finlayson et al., 2021; Soulos et al., 2020), and has been extended for use in mechanistic interpretability techniques which chain causes into graph structures. Variants include patching via adding Gaussian noise (also termed causal tracing; Meng et al., 2022). One can generalize patching to implicate multiple components simultaneously, as in path patching (Wang et al., 2023; Goldowsky-Dill et al., 2023). This technique has been applied extensively to understand mechanisms underlying sophisticated language model behaviors (Geva et al., 2023; Yu et al., 2023; Todd et al., 2024) and to edit language models (Meng et al., 2022; 2023).

Perhaps the most relevant methods to this work are those used to discover and analyze **circuits** (Wang et al., 2023; Conmy et al., 2023), which are subgraphs of a neural network that are causally implicated in performing a specific task. These generally employ activation patching, path patching, or some combination thereof. Circuits contain nested dependencies, but even if a node does not have a direct edge to the model output, it is often required that each circuit node directly influence the output when ablated. This may be thought of as a generalization of individual

[1] We believe this theoretically grounded definition of "mechanism" should be adopted in mechanistic interpretability. There do exist non-causal methods which are labeled as mechanistic interpretability, but even if the method does not rely on counterfactuals *per se*, seeking algorithmic understanding is essentially equivalent to seeking human-understandable causal mechanisms that concisely explain model behavior. If so, then even correlational methods can claim to pursue mechanistic understanding, albeit with lower faithfulness and causal efficacy on average than methods which causally implicate units of model computation. A benefit is that this definition may encourage more causally rigorous evaluations to verify claims of mechanistic understanding.

[2] Some use "counterfactual" to refer specifically to inputs where the answer is different from that given the original (often termed "clean") input. We believe this usage is too narrow: if a neuron is responsible for some behavior, and one manually sets its activation to some alternate value that flips, nullifies, or otherwise modifies its causal contribution, one will still observe an effect on the (probability of the) behavior in a manner that would not have been possible otherwise. We will use "patching ablation" to refer to the form of counterfactual based on inputs that flip the answer, but we will also refer to zero ablations and mean ablations as counterfactuals.

[3] There exist other interventions which do not require minimally different patch inputs or binary answers. For example, in the absence of a clear binary incorrect answer, one can also measure the negative logit of the correct completion. Rather than patching in a_{patch}, one could instead set a to its mean activation across a sample of general text (referred to as a **mean ablation**), or randomly sample an activation from an unrelated input (known as a **patching ablation**). A **zero ablation**, where a is replaced with 0, is generally not considered principled, unless one is working with units whose default values are 0 (as when working with sparse autoencoders; see Bricken et al. 2023 and Cunningham et al. 2024).

component probing, in that component probing searches for causal graphs of depth $= 1$, whereas circuit discovery locates causal graphs of depth > 1. Circuits are often used for post-hoc interpretability (Hanna et al., 2023; Lieberum et al., 2023; Chughtai et al., 2023), and sometimes to modify model behaviors via targeted ablations (Li et al., 2023; Marks et al., 2024). Another causally grounded line of work takes hypothesized causal graphs as inputs and aligns them with (potentially multidimensional) model subspaces, either via search or direct optimization over intervention site boundaries (Geiger et al., 2021; Wu et al., 2023; Geiger et al., 2024). Here, patching ablations are referred to as interchange interventions.

3. Counterfactual dependence is not transitive.

It is natural to assume that if C counterfactually depends on B, and B depends on A, then C must depend on A. This is a fallacy. Consider the following canonical example from Ned Hall:[4]

(1) A: A hiker is walking up a mountain. A large boulder begins rolling down the mountain toward the hiker.
B: The hiker, noticing the rolling boulder, ducks out of the way.
C: The hiker lives.

In this scenario, C depends on B, and B depends on A. Does it therefore make sense to say that the boulder rolling down the mountain caused the hiker to live? The reason this feels strange is because C is not counterfactually dependent on A: the hiker would have lived even if the boulder had never rolled down the mountain. Thus, while each local counterfactual dependence holds, the transitive dependence of C on A does not. In other words, **counterfactual dependence is not necessarily transitive**. See Figure 1 for an illustration.

However, causality intuitively feels transitive, and there are many cases where it is.[5] Consider playing a game of billiards, where one hits billiard ball A, which then hits ball B, which sinks ball C. In this case, C *does* counterfactually depend on A. What, then, is the fundamental difference between transitive and non-transitive dependency chains?

Halpern (2016) poses necessary and sufficient conditions

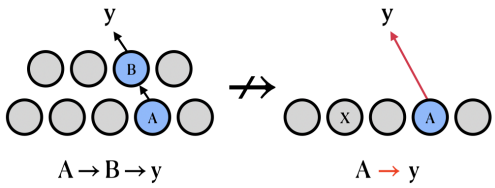

Figure 1. Illustration of the non-transitivity of counterfactual dependence. If an output y depends on B, and if B depends on A (potentially in addition to other nodes), then it does not necessarily entail that y depends on A.

to guarantee the transitivity of counterfactual dependence. Suppose there exist random variables A, B, and C. Let there exist values a_1 and a_2 that A could take, and there exist similar pairs of values b_1, b_2 and c_1, c_2 for B and C, respectively. Also let "$\rightarrow$" indicate entailment. Then the following conditions guarantee the transitive dependence of C on A:

1. $A = a_1 \rightarrow B = b_1$
2. $B = b_1 \rightarrow C = c_1$
3. $c_1 \neq c_2$
4. $A = a_2 \rightarrow B = b_2$
5. $A = a_2 \wedge B = b_2 \rightarrow C = c_2$

The intuition is that A can take values which determine at least two distinct values of B,[6] and that B can determine at least one value of C. Then, if *the conjunction of* A and B also determines a particular value of C that is distinct from the one that B alone caused, then the dependency chain from A to C is transitive, and C therefore counterfactually depends on A.

Halpern poses an additional set of sufficient (but not necessary) conditions to guarantee transitivity:

1. For every value b that B can take, there exists a value a that A can take such that $A = a \rightarrow B = b$.
2. B is in every causal path from A to C.

In neural networks, counterfactual dependencies will not often be transitive. While possible, it seems unlikely that either set of conditions would hold in most neural networks. The first is unlikely due to the existence of many neurons that simultaneously influence B, such that the probability is low of there existing some value a_1 of A that alone causes B to take some value b_1. The second is unlikely because, in the simplest case where B is in a hidden layer between A and C, there are n possible paths from a neuron in A's layer to a neuron in C's layer, where n is the number of neurons in B's layer. This number grows exponentially as we increase the layer distance between A and C. Thus, it is reasonable

[4] As reported in Hitchcock (2001) with credit to Hall.

[5] It has long been debated whether *actual causality*, as separate from counterfactual dependence, is always transitive (cf. Paul & Hall 2013; Lewis 1973; 1986, who claim that transitivity is a key desideratum of theories of causality). We are inclined to follow the analysis of Beckers & Vennekens (2017), who claim that the transitivity of causality is conditional, and who then give sufficient conditions for causal transitivity.

[6] This may be easier to conceptualize if one treats a_2, b_2, and c_2 as the *default values* of each variable, as Halpern does in leading up to this more general set of conditions.

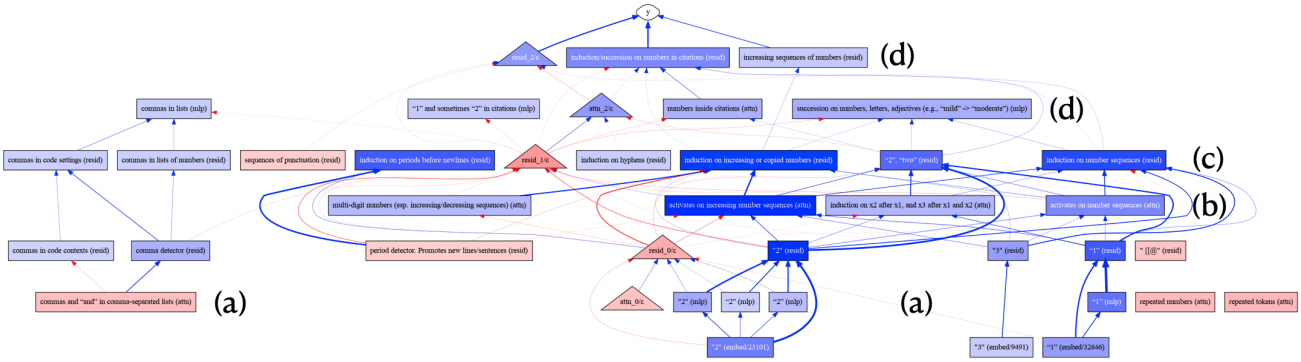

Figure 2. Sparse feature circuit for layer 3 attention feature 14579, a general succession feature (denoted y in the figure). Best viewed zoomed in. Darker blue shades indicate higher-magnitude positive effects, and darker red shades indicate higher-magnitude negative effects. Given inputs such as "1, 2, 3,", the model's first layers detect each number and the punctuation between them (a). The number detectors are inputs to number list detectors and increasing number sequence detectors (b). These features then inform induction features (c) that are only active in specific contexts or on particular classes of phenomena (like number sequences). These aid more general incrementor/succession features (d).

to assume by default that counterfactual dependencies will generally *not* be transitive in neural networks, unless B is some **causal bottleneck** through which information from node A *must* pass to arrive at C. There is no reason *a priori* to expect such bottlenecks to exist at the level of neurons, attention heads, or sparse autoencoder features, which is unfortunate given that these fine-grained units are more likely to be human-interpretable. One could view residuals as causal bottlenecks into which all information is collapsed at the end of a layer, so causal chains would be transitive if the intermediate node B were the entire residual *vector*; nonetheless, such coarse-grained units are generally difficult to interpret. Future work could consider methods for demonstrating the existence and frequency of interpretable causal bottlenecks in neural networks.

This presents a problematic trade-off. Methods that require components to significantly change the output will not recall components that explain how intermediate causal nodes arise. This hinders our understanding of how a model composes increasingly high-level intermediate abstractions to perform a task. However, if we employ methods which search over *all* local dependencies and do *not* require a significant change to the output, we will include many more nodes than is necessary to explain the model behavior, and probably significantly increase our method's time complexity.

3.1. Case study: Succession features in Pythia 70M

Circuits are directed acyclic graphs (DAGs), where each node represents a model component or set of components, and each edge represents a causal link from an upstream node to a downstream node. The leaf node is typically the target metric, such as the logit difference between a correct and a minimally different incorrect completion to some context. Circuits are intended to represent the subgraph of the model's computation graph that is causally implicated in increasing the target metric. To ensure that nodes in the circuit are human-interpretable, one can use sparse autoencoder dictionaries (Bricken et al., 2023; Cunningham et al., 2024) to decompose submodules (like MLP or attention layers) into human-interpretable sparse features.

In Marks et al. (2024), the authors discovered sparse feature circuits for various model behaviors, including a *specific succession* behavior where Pythia 70M predicts "4" given inputs containing numbered lists (where each number is potentially separated by variable-length text), as in "1: ...\n2: ...\n3: ...\n". After generating a sparse feature circuit for this seemingly coherent behavior, they find that it is actually the intersection of two distinct mechanisms: (1) a *general succession* mechanism (Gould et al., 2024), which promotes the next item in ordinal or numeric lists, and (2) a *specific induction* mechanism (Olsson et al., 2022), which copies previous elements of lists in the same context. As presented, the successor mechanism arises spontaneously at a layer 3 attention feature, which has no in edges; because local dependencies are not considered, *it is unclear how this succession abstraction is composed from the inputs*.

Thus, we obtain a sparse feature circuit (SFC) for cluster 382 from Marks et al. (2024), where the target metric y has been changed from the logit difference of two possible next tokens to the activation of the layer 3 attention succession feature (Figure 2). We use the same method as in Marks et al. (2024), changing only the target metric; see App. A for details and hyperparameters. We manually anno-

tate each feature by inspecting its activations and the logits that change most when the feature is ablated.[7] This SFC contains features and edges that were not in the original circuit, but which help explain how succession is computed in Pythia. For example, there is a causal chain from layer-0 features which detect specific numbers, to layer-1 features which detect sequences of numbers (regardless of order), to layer-2 features which detect increasing sequences of numbers. There is another mechanism where layer-0 features activate on all commas, then layer-1 features activate only on commas within lists of numbers, then layer-2 features activate on commas and elements of comma-separated number lists. This yields even stronger evidence that the target attention feature actually is a successor head—but also that it is primarily sensitive to *numeric* succession.

This is helpful for understanding the original feature circuit: the numeric successor feature that spontaneously arose in the original SFC is actually composed from the intersection of specific number detectors and number list detectors. This anecdote suggests that we should include some purely local dependencies if our goal is to fully understand how higher-level abstractions and decisions are composed from low-level input features. That said, it is easy to see how this would yield an exploding number of features if we include *all possible* local dependencies: the graph would become uninterpretable due to a rapidly increasing description length. For example, in the above SFC, we also found features which activate inside citations of the form "`[@<num>]`", where `<num>` is an integer. Is this useful? Maybe, but it may reflect how a particular input distribution is handled (this cluster contains a significant number of citations in this format) rather than how a generic behavior is composed. Another framing is that the citation feature is likely to reduce the description length of the feature less than the number list feature would, so we should prioritize explaining the number list features.

Given this discussion, should one keep local dependencies—i.e., upstream nodes that influence downstream nodes which affect the target metric, but not the target metric directly? Or should one instead mandate that each causal node we keep have some direct measurable effect on the target metric? **If one's goal is to understand how a model will generalize, one should also consider at least *some* local causal dependencies.** However, **if one's goal is merely to understand which components will directly affect downstream performance** (e.g., when editing or pruning models), **it may suffice to only include components that directly affect the output**.[8] This is fundamentally a trade-off between minimizing description length and increasing recall of highly explanative components, similar to the trade-off described in Sharkey (2024). In such cases, the best point in the Pareto frontier may simply consist of the minimal number of features needed to understand whether a model is making the right decisions *in the right way*.

To prevent the problem of overly long description lengths when including local dependencies, we recommend to (1) **use more diverse datasets when discovering and interpreting counterfactual dependencies**, and to (2) **directly evaluate the out-of-distribution generalization of interpretability methods by using held-out evaluation datasets that are distinct in controlled ways** from the dataset over which we discover dependencies. These will allow us to recover more causally relevant nodes while also helping us filter out those which are too input-specific or high-variance.

4. Redundant causes are systematically missed.

Counterfactual theories of causality struggle to handle **overdetermination**—i.e., cases where an effect has multiple distinct causes which are each sufficient on their own. Consider the following classic example from Hall (2004):

(2) A_1: Suzy throws a rock at a glass bottle.
A_2: At the same time, Billy throws a rock at the same bottle.
B: The rocks shatter the bottle.

If A_1 had not occurred, B still would have happened because of A_2, and vice versa. Thus, B is not counterfactually dependent on either A_1 or A_2 in isolation: ablating only one of them would produce no change to B. Does this mean that neither rock caused the bottle to break? This is an uncomfortable conclusion.

The above example feels contrived because simultaneous redundant causes of the same event are rare in the real world. However, in neural networks, redundancy is very common. Consider a layer with n neurons in the output vector of the MLP submodule: these are n simultaneous causes, at least some of which are likely to have similar effects on some downstream model component or on the model behavior. Indeed, redundancy is found to be very common in pre-trained neural networks (Dalvi et al., 2020), even for abstract phenomena like syntactic agreement (Tucker et al., 2022); further evidence comes from the fact that models can be distilled (Sanh et al., 2020) or pruned (Frantar & Alistarh, 2023) without significant loss in performance. This implies that **current interpretability methods that rely on ablations probably have significantly lower recall than expected, and it is unclear how many or what kinds of components are missing**.

[7] There are problems in assigning natural language explanations to model components. These are detailed in Huang et al. (2023).

[8] Though Hase et al. (2023) find that causal localizations do not accurately reflect the optimal locations to update parameters.

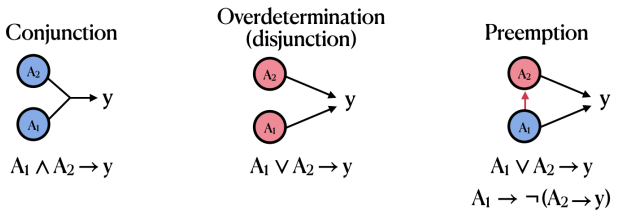

Figure 3. Illustration of the result of intervening on individual model components. The captions depict the actual causal graph, while the visualizations depict the causal graph that would have been discovered via ablations to individual components. Nodes in **blue** are in the actual causal graph and would be discoverable. Nodes in **pink** are in the actual causal graph, but would not have been discovered. Black edges are cause-effect relationships, whereas red edges indicate preemption.

Suppose for example that we have two neurons A_1 and A_2 which have activations a_1 and a_2, respectively, given input x. $A_1 = a_1$ and $A_2 = a_2$ cause the activation of B to increase. In the absence of the other, both $A_1 = a_1$ and $A_2 = a_2$ increase B to some value close to b, which causes model behavior y. But if both are active, then suppose that the increase in B's activation beyond b is non-linear and small—e.g., it could be logarithmic or at the upper plateau of a logistic function. Thus, in a search over individual model components, intervening on A_1 or A_2 will have a small marginal effect on B. We might therefore be misled into concluding that *neither* A_1 *nor* A_2 causally contribute to B, and thus do not cause the target behavior. See Figure 3 for an illustration. This applies straightforwardly to cases where two neurons have strongly correlated effects, even if their functional roles are not the same. As a less trivial example, assume we are searching for components responsible for successfully performing a question answering task. If A_1 detects geography terms, A_2 detects political terms, and B predicts country names, then A_1 and A_2 may have very similar effects on B on certain narrow distributions, despite being sensitive to different phenomena. We might be able to find them if A_1 and A_2 affect other components or metrics (e.g., examples of opposite labels or from different domains) in distinct and measurable ways. This implies that **diverse input data and the choice of metric will be crucial to discovering all relevant components**.

There also exist less trivial forms of redundant causation that have already been attested in neural networks. Backup heads (Wang et al., 2023) and Hydra effects (McGrath et al., 2023) are examples of what is called **preemption** in the causality literature. Returning to Ex. (2), if Suzy's rock A_1 had hit the bottle first, then Billy's rock A_2 would not have shattered the bottle B—but preventing A_1 would still not prevent B due to A_2. See Figure 3 for an illustration. In preemption, A_1 and A_2 are sufficient on their own to cause y. Additionally, when A_1 is active, it triggers some mechanism that causes A_2's marginal effect on y to be nullified. When A_1 is ablated, A_2 will cause y. Here, we would discover A_1 through simple counterfactual ablations, but we would not discover A_2 without first ablating A_1.

Probing individual model components is problematic. The implications of these phenomena are far-reaching: overdetermination and preemption affect not only circuit discovery methods, but also *any* intervention method where we have many possible simultaneous causes. If one does not test all possible combinations of model components, then one is not guaranteed to recall the full mechanism leading to some model behavior. It may be easier to discover cases of preemption, since one can ablate components that have already been discovered and observe whether new components then affect the output. With overdetermination, it is not clear where to search for these component sets; thus, some type of search over sets of components—rather than individual components—may be necessary.

Given minimally differing input/output pairs, Heimersheim & Nanda (2024) advocate overcoming this problem without multi-component ablations by using multiple types of patching ablations: one as described in §2.1, and another where x_{original} and x_{patch} are swapped (termed "noising" and "denoising" ablations, respectively). We agree, and to generalize this beyond minimal pairs, we recommend **using both positive and negative counterfactuals when discovering causal dependencies**—i.e., do not just ablate a component when it is active, but also inject high activations into a component when it is not active. This corresponds to a search where a causal node can be included in the graph if either (i) ablating the node has a negative effect on the behavior, or (ii) injecting it causes the behavior to occur.

5. Discussion

These challenges pose the following questions for interpretability researchers who employ counterfactuals:

1. What should we do with nodes that are locally important, but have little effect on the output?
2. How can we increase recall without significantly harming precision?
3. Are counterfactual theories of causality sufficient for interpreting neural networks?

1. Keep at least some local dependencies. On (1), one should interpret each edge in a causal graph—such as a circuit edge or a connection between aligned subspaces—as a purely local causal relationship by default.[9] One should remain agnostic about the relationship between a given pair

[9]Unless all edges represent influence on the output, as in Marks et al. (2024).

of nodes A and B that are not directly connected in the graph—unless the node(s) between them are a causal bottleneck and/or can be determined by A, in which case one may assume that B is directly counterfactually dependent on A. To ensure that the nodes in a circuit actually affect the outputs, one might respond by testing the effect of a given node on the output metric and only including it if it directly affects the logits. This poses its own problems: just as it did not make sense to talk about the hiker ducking and then surviving without first mentioning the boulder rolling down the mountain in Ex. (1), it would similarly not make sense to exclude all local causal relationships where a node does not affect the output directly. Indeed, this would be a catastrophic omission when telling the full causal story of how certain intermediate representations which *are* important are composed from simpler abstractions. We therefore advocate to **keep and analyze components that do not directly affect the outputs by themselves**, but that *do* causally effect other downstream nodes in a circuit.

To prevent the opposite problem of including too many nodes and decreasing precision, we should prioritize including information which is most likely to reduce the description length of the program that implements the target behavior. One way to do this is by **deploying more diverse datasets**, such that components which only explain a small subset of inputs or minor aspects of the behavior would be less likely to pass our effect size thresholds. We should use input datasets and controlled evaluation datasets which are neither confined to single domains nor highly templatic.

The above discussion focuses on cases where mediators are neurons or attention heads within a single layer. This becomes more confusing when intervening on multiple components: if path patching is used, all nodes within the path should be considered as a single node/mediator. Effects within a path p_1 on y will be transitive, but effects between path p_1 on the output y via p_2 may not be transitive unless p_1 and p_2 satisfy a set of conditions from §3. This raises another recommendation: if you employ interventions, **be explicit about what your causal nodes/mediators are, especially if you are drawing connections between them.**

2. Robust methods will *require* a diversity of interventions and/or interventions to combinations of components. On (2), the naïve brute-force solution would be to intervene on all possible combinations of components. Of course, it is not tractable to test all possible combinations of components, even when employing efficient approximations like attribution patching (Syed et al., 2023; Kramár et al., 2024). A reasonable compromise may be to rely on greedy search methods (as in §4.2 of Vig et al., 2020).

More tractable methods for capturing overdetermined causes will probably entail a diversity of data and intervention types. We can, for example, use both positive and negative counterfactuals (as described in §4). As a starting point, we could construct shared toy models which have overdetermination and preemption built in; using these, we can develop methods that can tractably discover redundant causes. These methods can later be scaled to larger and more capable neural models.

3. Counterfactual theories are sufficient—but we must interpret them correctly. On (3): if counterfactual theories do not handle these issues well, perhaps there exist better theories. It would not suffice to return to **regularity theories** of causality (Mill, 1846), which ultimately reduce to equating correlation and causation, and therefore reduce precision via the inclusion of false positives. **Inferential theories** propose rules under which certain causes are *always* followed by specific effects (Mackie, 1965). There have been significant advances in inferential theories that explicitly tackle overdetermination and non-transitivity (e.g., Andreas & Günther, 2021; 2024), but despite this, it would be difficult to obtain satisfying empirical evidence of causal relationships as distinct from strong correlations.

An appealing alternative theory could be **causality as a transfer of conserved quantitities** (Salmon, 1984; 1994): if cause A transfers some conserved quantity to B in the interval of time $(A, B]$, and the same overall quantity remains in $A \vee B$ throughout that time, then A has causal influence on B. In LSTMs (Hochreiter & Schmidhuber, 1997) or state space models (Gu et al., 2021; 2022), this quantity could be related to the state vector carried across time steps; causal analyses could therefore be framed in terms of components which write to and erase from this state.[10] In a Transformer network (Vaswani et al., 2017), what might this conserved quantity be? There is also **causality as Kolmogorov complexity reduction** (Alexander & Gilboa, 2023; Janzing & Schölkopf, 2010), where the inclusion of cause C reduces the minimum description length of the effect E. However, this merely captures variables which sufficiently *describe* the effect, and which do not necessarily *execute* it. Further, this theory has been proposed primarily to capture the human notion of subjective causality, rather than actual causality.

Importantly, many non-counterfactual theories are not necessarily motivated by counterfactuals being inaccurate approximations of causality; rather, many are motivated by counterfactuals being difficult to obtain in observational settings, as in clinical trials. Fortunately, in neural networks, counterfactuals are easy to obtain: we have complete control over computation graphs and full access to the values of and

[10] This is similar to ideas proposed in Ferrando & Voita (2024) for Transformers, though they use similarities as a proxy for which components were important to a downstream submodule output, which is not guaranteed to capture faithful causal subgraphs.

connections between each node in the graph. Thus, despite its challenges, counterfactual theories are still likely the most actionable for interpretability research based on neural networks. Even without access to counterfactual *inputs*, the activation of a particular node can simply be nulled out via setting it to its mean activation, or sampling an activation from an unrelated context. While not fully principled,[11] this is still a tractable way to establish a sufficient condition for causal dependence.

In short, the challenge is not that the counterfactual theory has problems: the challenge is that these problems are difficult—but possible—to tractably address at scale. As a starting point, future work could proceed by creating toy models where overdetermination, preemption, and non-transitive dependencies are built into the network. This would allow us to design methods that can adequately handle the problems outlined here; later, these could be made more scalable via approximate metrics or heuristic-based search methods.

5.1. We need better causal mediators.

Much of this paper has assumed that, given access to a causal graph, one can interpret the mediators/nodes in the graph. This is often not the case. Methods where the nodes *can* be interpreted tend to be less flexible and open-ended due to the required effort and compute: for example, sparse feature circuits (Marks et al., 2024) require significant compute for training autoencoders, and these autoencoders may not generalize to a model that has been significantly tuned, pruned, and/or edited. That said, this is a one-time cost: once these autoencoders have been trained, they can be released and easily deployed for others to use. In distributed alignment search (Wu et al., 2023; Geiger et al., 2024), we are required to have a prior hypothesis as to how a model accomplishes a task, and the evaluation metrics do not tell us in what ways our hypotheses are wrong; thus, refining hypotheses may be non-trivial when a model implements a complex behavior in a non-human-like way.

On the other hand, methods that are more flexible and that do not require prior hypotheses often yield causal graphs with uninterpretable nodes. Thus, more work is needed on producing Pareto improvements with respect to human interpretability and compute-/labor-efficiency. Sparse autoencoder dictionaries represented a push in a more fine-grained direction, but other methods could consider learning more coarse and high-level abstractions—e.g., high-dimensional structures or supersets of components which have a coherent functional role.

[11] Consider this example from Nanda (2023): a pair of neurons' activations may be uniformly distributed at points along a circle; here, the mean activation of the pair would be the center of the circle, which results in activations that the neurons would never actually have on any input.

5.2. Possible Counterarguments

Possible counterargument 1: "Sacrificing a small amount of recall is acceptable." We anticipate one common counterargument to be that sacrificing some recall is acceptable when the search space of mediators (and combinations thereof) is large. We would argue against this as a general recommendation: there may exist classes of phenomena that are more likely to be encoded redundantly than others. This means that systematically excluding overdeterminers may bias our findings in a way that hides particular types of features more than others. Thus, if one aims to obtain the most accurate and complete picture of how a model accomplishes a behavior, then one must capture redundant causes, as well as mediators that primarily affect downstream mediators and not the output metric.

Possible counterargument 2: "These problems primarily apply to binary notions of causality. Neural networks are continuous." Another concern is that the above examples have depended on binary accounts of causality, where an event either causes an effect or it does not. Indeed, neural networks are much more forgiving: they may be interpreted as *probabilistic* or *continuous* causal graphs, where edges between nodes are weighted and where effects may be measured via continuous metrics like indirect effects. Current interpretability methods already rely on such metrics. And indeed, Lewis (2000) defines causality as whether a cause counterfactually *influences* the manner in which an effect occurs, rather than as a binary notion of whether E happens. This means that causes can have varying degrees and types of influence on a given effect, which makes partial causes easier to handle.

Nonetheless, the same non-transitivity and overdetermination issues still apply. The issue has simply shifted: *indirect effects are not transitively conserved in causal chains*. For example, given a neuron A that strongly influences the activation of neuron B, and neuron B that strongly influences output y, A may still have a very small effect on y. A might trigger or frequently co-occur with another mechanism that nullifies B's effect on y. And sets of variables $A_{1,2,\ldots,n}$ that result in B taking similar values would still be difficult to completely recall when using interventions to just one of these variables at a time.

Moreover, causal interpretability methods often define some arbitrary effect size threshold above which we keep some component in a circuit, or where we consider the component to have a significant enough contributition to the output phenomenon to be considered causally relevant. In other words, we start with continuous effect sizes, but then collapse our final results into discrete subsets or subgraphs of

components that are considered causes, while excluded components are not considered causes. This effectively gives us binary causal graphs, which makes structures such as circuits even more easily susceptible to the same fallacies described here—at least in their current presentation.

6. Broader Impact and Limitations

This paper discusses counterintuitive challenges that arise under counterfactual theory of causality. These challenges could partially explain why methods that apply or edit discovered components often do not generalize robustly. For example, in model editing, could this explain in part why the most causally relevant components discovered via counterfactual interventions are not the most effective locations to edit (Hase et al., 2023)? Additionally, given that information is gradually accumulated across layers (Meng et al., 2022; 2023; Geva et al., 2021; 2022), might the inclusion of locally but not necessarily globally relevant causal dependencies improve the faithfulness of the causal subgraphs we extract?

We acknowledge that this analysis could be improved or expanded in various ways. While the problems discussed here should impact any method that relies on counterfactual interventions to a subset of neural network components, this paper is written primarily with circuit discovery (Elhage et al., 2021; Conmy et al., 2023), alignment search (Geiger et al., 2021), or component set discovery methods (Vig et al., 2020) in mind. This is not a representative sample of causal and mechanistic interpretability methods. For example, it is not immediately intuitive how these problems affect methods that derive projections to erase or guard particular concepts in latent space (Belrose et al., 2023; Ravfogel et al., 2020; Elazar et al., 2021), or methods that leverage gradients from parametric classifiers to update the model at specific token positions (e.g., Giulianelli et al., 2018). One could argue that learned projections or modifications to a single layer will still be susceptible to preemption and non-transitivity concerns, since locally relevant early information and preempted later information may be missed by interventions that only affect the model at a single layer; nonetheless, if this layer is a causal bottleneck, non-transitivity may not apply if the entire vector is modified. This should be explored in more detail in future work.

Additionally, simultaneous overdetermination has not yet been attested in large language models. While there are already examples of preemption (Wang et al., 2023; McGrath et al., 2023), demonstrations that overdetermination is possible in language models (Heimersheim & Nanda, 2024), and demonstrations that redundancy is common (Tucker et al., 2022), it is not clear whether there exist groups of components in large language models that have the same effect on some downstream component or output behavior when either some subset or all such components in the set are active. Generally, when multiple causes trigger the same effect in a neural network, it will often be *additive*, rather than truly *redundant*: i.e., each component has a measurable marginal contribution to the magnitude of the effect, which makes each component discoverable. Thus, it is still not clear in what settings we would miss simultaneous causes, or in which settings they are most likely to arise.

7. Conclusion

We have described two problems for interpretability researchers who base their methods on counterfactual interventions: non-transitivity and overdetermination. The former complicates methods for discovering and interpreting deep causal graphs, while the latter lowers recall in systematic ways. We have discussed the implications of these findings when interpreting findings from mechanistic interpretability studies, and proposed suggestions for future work in this space.

Acknowledgments

We are grateful to Eric Todd, Can Rager, Yonatan Belinkov, and David Bau for providing feedback on an earlier version of this work. We are also thankful to the reviewers of the 2024 ICML Workshop on Mechanistic Interpretability for their helpful comments. This work was produced while A.M. was funded by a postdoctoral fellowship under the Zuckerman STEM Leadership program.

References

Alexander, Y. and Gilboa, I. Subjective causality. *Revue économique*, 74(4):619–633, 2023.

Andreas, H. and Günther, M. A Ramsey test analysis of causation for causal models. *The British Journal for the Philosophy of Science*, 2021.

Andreas, H. and Günther, M. A regularity theory of causation. *Pacific Philosophical Quarterly*, 2024.

Andreas, J. Tweet in response to Alexander Rush on the rise of mechanistic interpretability, 2024. URL https://twitter.com/jacobandreas/status/1749917888230883566.

Beckers, S. and Vennekens, J. The transitivity and asymmetry of actual causation. *Ergo: An Open Access Journal of Philosophy*, 4:1–27, 2017.

Belrose, N., Schneider-Joseph, D., Ravfogel, S., Cotterell, R., Raff, E., and Biderman, S. LEACE: Perfect linear concept erasure in closed form. In *Thirty-seventh Conference on Neural Information Processing Systems*,

2023. URL https://openreview.net/forum?id=awIpKpwTwF.

Biderman, S., Schoelkopf, H., Anthony, Q. G., Bradley, H., O'Brien, K., Hallahan, E., Khan, M. A., Purohit, S., Prashanth, U. S., Raff, E., et al. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, pp. 2397–2430. PMLR, 2023.

Bricken, T., Templeton, A., Batson, J., Chen, B., Jermyn, A., Conerly, T., Turner, N., Anil, C., Denison, C., Askell, A., Lasenby, R., Wu, Y., Kravec, S., Schiefer, N., Maxwell, T., Joseph, N., Hatfield-Dodds, Z., Tamkin, A., Nguyen, K., McLean, B., Burke, J. E., Hume, T., Carter, S., Henighan, T., and Olah, C. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. URL https://transformer-circuits.pub/2023/monosemantic-features/index.html.

Chughtai, B., Chan, L., and Nanda, N. A toy model of universality: Reverse engineering how networks learn group operations. In *Proceedings of the 40th International Conference on Machine Learning*. JMLR, 2023.

Conmy, A., Mavor-Parker, A. N., Lynch, A., Heimersheim, S., and Garriga-Alonso, A. Towards automated circuit discovery for mechanistic interpretability. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Cunningham, H., Ewart, A., Riggs, L., Huben, R., and Sharkey, L. Sparse autoencoders find highly interpretable features in language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=F76bwRSLeK.

Dalvi, F., Sajjad, H., Durrani, N., and Belinkov, Y. Analyzing redundancy in pretrained transformer models. In Webber, B., Cohn, T., He, Y., and Liu, Y. (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 4908–4926, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.398. URL https://aclanthology.org/2020.emnlp-main.398.

Elazar, Y., Ravfogel, S., Jacovi, A., and Goldberg, Y. Amnesic Probing: Behavioral Explanation with Amnesic Counterfactuals. *Transactions of the Association for Computational Linguistics*, 9:160–175, 03 2021. ISSN 2307-387X. doi: 10.1162/tacl_a_00359. URL https://doi.org/10.1162/tacl_a_00359.

Elhage, N., Nanda, N., Olsson, C., Henighan, T., Joseph, N., Mann, B., Askell, A., Bai, Y., Chen, A., Conerly, T., DasSarma, N., Drain, D., Ganguli, D., Hatfield-Dodds, Z., Hernandez, D., Jones, A., Kernion, J., Lovitt, L., Ndousse, K., Amodei, D., Brown, T., Clark, J., Kaplan, J., McCandlish, S., and Olah, C. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. URL https://transformer-circuits.pub/2021/framework/index.html.

Ferrando, J. and Voita, E. Information flow routes: Automatically interpreting language models at scale. *Computing Research Repository*, arXiv:2403.00824, 2024. URL https://arxiv.org/abs/2403.00824.

Finlayson, M., Mueller, A., Gehrmann, S., Shieber, S., Linzen, T., and Belinkov, Y. Causal analysis of syntactic agreement mechanisms in neural language models. In Zong, C., Xia, F., Li, W., and Navigli, R. (eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 1828–1843, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.144. URL https://aclanthology.org/2021.acl-long.144.

Frantar, E. and Alistarh, D. SparseGPT: massive language models can be accurately pruned in one-shot. In *Proceedings of the 40th International Conference on Machine Learning*, 2023 International Conference on Machine Learning. Journal of Machine Learning Research, 2023.

Geiger, A., Lu, H., Icard, T. F., and Potts, C. Causal abstractions of neural networks. In Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=RmuXDtjDhG.

Geiger, A., Wu, Z., Potts, C., Icard, T., and Goodman, N. D. Finding alignments between interpretable causal variables and distributed neural representations. *Computing Research Repository*, arXiv:2303.02536, 2024. URL https://arxiv.org/abs/2303.02536.

Geva, M., Schuster, R., Berant, J., and Levy, O. Transformer feed-forward layers are key-value memories. In Moens, M.-F., Huang, X., Specia, L., and Yih, S. W.-t. (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 5484–5495, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.446. URL https://aclanthology.org/2021.emnlp-main.446.

Geva, M., Caciularu, A., Wang, K., and Goldberg, Y. Transformer feed-forward layers build predictions by promot-

ing concepts in the vocabulary space. In Goldberg, Y., Kozareva, Z., and Zhang, Y. (eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 30–45, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.3. URL https://aclanthology.org/2022.emnlp-main.3.

Geva, M., Bastings, J., Filippova, K., and Globerson, A. Dissecting recall of factual associations in auto-regressive language models. In Bouamor, H., Pino, J., and Bali, K. (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 12216–12235, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.751. URL https://aclanthology.org/2023.emnlp-main.751.

Giulianelli, M., Harding, J., Mohnert, F., Hupkes, D., and Zuidema, W. Under the hood: Using diagnostic classifiers to investigate and improve how language models track agreement information. In Linzen, T., Chrupała, G., and Alishahi, A. (eds.), *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 240–248, Brussels, Belgium, November 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-5426. URL https://aclanthology.org/W18-5426.

Goldberger, A. S. Structural equation methods in the social sciences. *Econometrica*, 40(6):979–1001, 1972. ISSN 00129682, 14680262. URL http://www.jstor.org/stable/1913851.

Goldowsky-Dill, N., MacLeod, C., Sato, L., and Arora, A. Localizing model behavior with path patching. *Computing Research Repository*, arXiv:2304.05969, 2023. URL https://arxiv.org/abs/2304.05969.

Gould, R., Ong, E., Ogden, G., and Conmy, A. Successor heads: Recurring, interpretable attention heads in the wild. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=kvcbV8KQsi.

Gu, A., Johnson, I., Goel, K., Saab, K. K., Dao, T., Rudra, A., and Re, C. Combining recurrent, convolutional, and continuous-time models with linear state space layers. In Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=yWd42CWN3c.

Gu, A., Goel, K., and Re, C. Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=uYLFoz1vlAC.

Hall, N. Two concepts of causation. In Collins, J., Hall, N., and Paul, L. (eds.), *Causation and Counterfactuals*, pp. 225–276. MIT Press, 2004.

Halpern, J. Y. Sufficient conditions for causality to be transitive. *Philosophy of Science*, 83(2):213–226, 2016.

Hanna, M., Liu, O., and Variengien, A. How does GPT-2 compute greater-than?: Interpreting mathematical abilities in a pre-trained language model. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=p4PckNQR8k.

Hase, P., Bansal, M., Kim, B., and Ghandeharioun, A. Does localization inform editing? Surprising differences in causality-based localization vs. knowledge editing in language models. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=EldbUlZtbd.

Heimersheim, S. and Nanda, N. How to use and interpret activation patching. *Computing Research Repository*, arXiv:2404.15255, 2024. URL https://arxiv.org/abs/2404.15255.

Hitchcock, C. The intransitivity of causation revealed in equations and graphs. *The Journal of Philosophy*, 98(6):273–299, 2001. ISSN 0022362X. URL http://www.jstor.org/stable/2678432.

Hochreiter, S. and Schmidhuber, J. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, nov 1997. ISSN 0899-7667. doi: 10.1162/neco.1997.9.8.1735. URL https://doi.org/10.1162/neco.1997.9.8.1735.

Huang, J., Geiger, A., D'Oosterlinck, K., Wu, Z., and Potts, C. Rigorously assessing natural language explanations of neurons. In Belinkov, Y., Hao, S., Jumelet, J., Kim, N., McCarthy, A., and Mohebbi, H. (eds.), *Proceedings of the 6th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pp. 317–331, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.blackboxnlp-1.24. URL https://aclanthology.org/2023.blackboxnlp-1.24.

Hume, D. *An Enquiry Concerning Human Understanding*. 1748.

Janzing, D. and Schölkopf, B. Causal inference using the algorithmic markov condition. *IEEE Transactions on Information Theory*, 56(10):5168–5194, 2010.

Kramár, J., Lieberum, T., Shah, R., and Nanda, N. Atp*: An efficient and scalable method for localizing llm behaviour to components. *Computing Research Repository*, arXiv:2403.00745, 2024. URL https://arxiv.org/abs/2403.00745.

Lewis, D. Causation. In Crane, T. and Farkas, K. (eds.), *Philosophical Papers II*, pp. 159–213. Oxford University Press, 1986.

Lewis, D. Causation as influence. *The Journal of Philosophy*, 97(4):182–197, 2000. ISSN 0022362X. URL http://www.jstor.org/stable/2678389.

Lewis, D. K. *Counterfactuals*. Blackwell, Malden, Massachusetts, 1973.

Li, M., Davies, X., and Nadeau, M. Circuit breaking: Removing model behaviors with targeted ablation. In *Workshop on Challenges in Deployable Generative AI at the International Conference on Machine Learning (ICML)*, 2023. URL https://arxiv.org/abs/2309.05973.

Lieberum, T., Rahtz, M., Kramár, J., Nanda, N., Irving, G., Shah, R., and Mikulik, V. Does circuit analysis interpretability scale? evidence from multiple choice capabilities in chinchilla. *Computing Research Repository*, arXiv:2307.09458, 2023. URL https://arxiv.org/abs/2307.09458.

Mackie, J. L. Causes and conditions. *American Philosophical Quarterly*, 2(4):245–264, 1965.

Marks, S., Rager, C., Michaud, E. J., Belinkov, Y., Bau, D., and Mueller, A. Sparse feature circuits: Discovering and editing interpretable causal graphs in language models. *Computing Research Repository*, arXiv:2403.19647, 2024. URL https://arxiv.org/abs/2403.19647.

McGrath, T., Rahtz, M., Kramar, J., Mikulik, V., and Legg, S. The hydra effect: Emergent self-repair in language model computations. *Computing Research Repository*, arXiv:2307.15771, 2023. URL https://arxiv.org/abs/2307.15771.

Meng, K., Bau, D., Andonian, A., and Belinkov, Y. Locating and editing factual associations in GPT. *Advances in Neural Information Processing Systems*, 35, 2022.

Meng, K., Sen Sharma, A., Andonian, A., Belinkov, Y., and Bau, D. Mass editing memory in a transformer. *The Eleventh International Conference on Learning Representations (ICLR)*, 2023.

Mill, J. S. *A System of Logic, Ratiocinative and Inductive: Being a Connected View of the Principles of Evidence, and the Methods of Scientific Investigation*. Cambridge University Press, London, England, 1846.

Nanda, N. A comprehensive mechanistic interpretability explainer & glossary. *Personal blog*, 2023. URL https://www.neelnanda.io/mechanistic-interpretability/glossary.

Olsson, C., Elhage, N., Nanda, N., Joseph, N., DasSarma, N., Henighan, T., Mann, B., Askell, A., Bai, Y., Chen, A., Conerly, T., Drain, D., Ganguli, D., Hatfield-Dodds, Z., Hernandez, D., Johnston, S., Jones, A., Kernion, J., Lovitt, L., Ndousse, K., Amodei, D., Brown, T., Clark, J., Kaplan, J., McCandlish, S., and Olah, C. In-context learning and induction heads. *Transformer Circuits Thread*, 2022. URL https://transformer-circuits.pub/2022/in-context-learning-and-induction-heads/index.html.

Paul, L. A. and Hall, N. *Causation: A User's Guide*. Oxford University Press UK, Oxford, 2013.

Pearl, J. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2000.

Pearl, J. Direct and indirect effects. In *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence*, pp. 411–420. Morgan Kaufmann, 2001.

Ravfogel, S., Elazar, Y., Gonen, H., Twiton, M., and Goldberg, Y. Null it out: Guarding protected attributes by iterative nullspace projection. In Jurafsky, D., Chai, J., Schluter, N., and Tetreault, J. (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 7237–7256, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.647. URL https://aclanthology.org/2020.acl-main.647.

Robins, J. M. and Greenland, S. Identifiability and exchangeability for direct and indirect effects. *Epidemiology*, 3(2):143–155, 1992. ISSN 10443983. URL http://www.jstor.org/stable/3702894.

Salmon, W. C. *Scientific Explanation and the Causal Structure of the World*. Princeton University Press, 1984.

Salmon, W. C. Causality without counterfactuals. *Philosophy of Science*, 61(2):297–312, 1994. ISSN 00318248, 1539767X. URL http://www.jstor.org/stable/188214.

Sanh, V., Debut, L., Chaumond, J., and Wolf, T. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *Computing Research Repository*, arXiv:1910.01108, 2020. URL https://arxiv.org/abs/1910.01108.

Sharkey, L. Sparsify: A mechanistic interpretability research agenda, 2024. URL https://www.lesswrong.com/posts/64MizJXzyvrYpeKqm/sparsify-a-mechanistic-interpretability-research-agenda.

Soulos, P., McCoy, R. T., Linzen, T., and Smolensky, P. Discovering the compositional structure of vector representations with role learning networks. In Alishahi, A., Belinkov, Y., Chrupała, G., Hupkes, D., Pinter, Y., and Sajjad, H. (eds.), *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pp. 238–254, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.blackboxnlp-1.23. URL https://aclanthology.org/2020.blackboxnlp-1.23.

Sundararajan, M., Taly, A., and Yan, Q. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pp. 3319–3328. JMLR.org, 2017.

Syed, A., Rager, C., and Conmy, A. Attribution patching outperforms automated circuit discovery. In *NeurIPS Workshop on Attributing Model Behavior at Scale*, 2023. URL https://openreview.net/forum?id=tiLbFR4bJW.

Todd, E., Li, M. L., Sharma, A. S., Mueller, A., Wallace, B. C., and Bau, D. Function vectors in large language models. In *Proceedings of the 2024 International Conference on Learning Representations*, 2024.

Tucker, M., Eisape, T., Qian, P., Levy, R., and Shah, J. When does syntax mediate neural language model performance? evidence from dropout probes. In Carpuat, M., de Marneffe, M.-C., and Meza Ruiz, I. V. (eds.), *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 5393–5408, Seattle, United States, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.naacl-main.394. URL https://aclanthology.org/2022.naacl-main.394.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Vig, J., Gehrmann, S., Belinkov, Y., Qian, S., Nevo, D., Singer, Y., and Shieber, S. Investigating gender bias in language models using causal mediation analysis. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 12388–12401. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/92650b2e92217715fe312e6fa7b90d82-Paper.pdf.

Wang, K. R., Variengien, A., Conmy, A., Shlegeris, B., and Steinhardt, J. 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.

Wu, Z., Geiger, A., Icard, T., Potts, C., and Goodman, N. Interpretability at scale: Identifying causal mechanisms in alpaca. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=nRfClnMhVX.

Yu, Q., Merullo, J., and Pavlick, E. Characterizing mechanisms for factual recall in language models. In Bouamor, H., Pino, J., and Bali, K. (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 9924–9959, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.615. URL https://aclanthology.org/2023.emnlp-main.615.

A. Discovering a Sparse Feature Circuit for a Sparse Feature

In §3, we discovered a sparse feature circuit containing components that cause a downstream sparse feature to activate at higher positive values. This circuit was discovered with Pythia 70M (Biderman et al., 2023) using the sparse autoencoder dictionaries released with Marks et al. (2024).[12] We use the code and data released by the authors; the dataset for this succession cluster is from `https://feature-circuits.xyz`, and corresponds to Cluster 382 in Marks et al. (2024).

The primary difference between this sparse feature circuit and the original circuit for this cluster is the target metric. In the original work, the metric was defined as the negative log-probability of the token x_{t+1} given some context $x_{1..t}$:

$$y = -\log p_\theta(x_{t+1}|x_1, \ldots, x_t)$$

In §3, the target metric is the negative activation of the layer 3 attention feature at index 14579. Here, we use "feature" to refer to a single index of a sparse autoencoder's encoded latent space. Specifically, given the output vector $\mathbf{x}$ of the out projection of the layer 3 attention block, the succession feature is index 14579 of the vector $\mathbf{f}$ defined by the encoder

$$\mathbf{f} = \text{ReLU}(W_e(\mathbf{x} - \mathbf{b}_d) + \mathbf{b}_e)$$

This operation represents the encoder of the sparse autoencoder dictionary; W_e, $\mathbf{b}_d$, and $\mathbf{b}_e$ are learned terms. Then, we simply set $y = -\mathbf{f}_{14579}$.

A feature is kept in the circuit if its indirect effect (IE) on y surpasses node threshold T_N, a hyperparameter. The IE is computed by ablating the activation of upstream sparse feature $\mathbf{a}$, and then measuring the change in y; this change should be positive if the node causally contributes to y. In practice, to compute the IE, one must run $O(n)$ forward passes, where n is the number of components; given that there are nearly 600,000 sparse features to search over, this is not tractable nor scalable. Thus, a linear approximation of the indirect effect $\hat{\text{IE}}$ is used, as it requires only two forward passes and one backward pass ($O(1)$ passes) for all model components:

$$\hat{\text{IE}} = \frac{\partial y}{\partial \mathbf{a}}\bigg|_{\mathbf{a}=\mathbf{a}_{\text{original}}} \cdot (\mathbf{a}_{\text{patch}} - \mathbf{a}_{\text{original}})$$

Here, $\frac{\partial y}{\partial \mathbf{a}}\Big|_{\mathbf{a}=\mathbf{a}_{\text{original}}}$ is the gradient of the sparse feature $\mathbf{a}$ given the original input, whereas $\mathbf{a}_{\text{patch}}$ and $\mathbf{a}_{\text{original}}$ are the activation of the feature given the patch input and original input, respectively. For this cluster circuit, there is no patch input; therefore, we set $\mathbf{a}_{\text{patch}} = 0$, often called a zero ablation. Note that zero ablations, while not principled for neurons (whose default values may not be 0 and where differences in sign are not necessarily meaningful), sparse autoencoders are trained such that any given feature's activation should be zero on most inputs, and only take positive activations when the feature contributes to $\mathbf{x}$ taking different values than its mean. To make this more accurate, a slightly more expensive and more accurate approximation based on integrated gradients (Sundararajan et al., 2017) is used; see Marks et al. (2024) for details.

In these experiments, the node threshold T_N is set to 0.4 and the edge threshold T_E is set to 0.04. The circuit was discovered using a single A6000 GPU.

[12] Released at `https://github.com/saprmarks/dictionary_learning`.