How Reliable are Causal Probing Interventions?

Marc E. Canby* **Adam Davies*** **Chirag Rastogi** **Julia Hockenmaier**
Siebel School of Computing and Data Science
The Grainger College of Engineering
University of Illinois Urbana-Champaign
{marcec2,adavies4,chiragr2,juliahmr}@illinois.edu

Abstract

Causal probing aims to analyze foundation models by examining how intervening on their representation of various latent properties impacts their outputs. Recent works have cast doubt on the theoretical basis of several leading causal probing methods, but it has been unclear how to systematically evaluate the effectiveness of these methods in practice. To address this, we define two key causal probing desiderata: *completeness* (how thoroughly the representation of the target property has been transformed) and *selectivity* (how little non-targeted properties have been impacted). We find that there is an inherent tradeoff between the two, which we define as *reliability*, their harmonic mean. We introduce an empirical analysis framework to measure and evaluate these quantities, allowing us to make the first direct comparisons between different families of leading causal probing methods (e.g., linear vs. nonlinear, or concept removal vs. counterfactual interventions). We find that: (1) all methods show a clear tradeoff between completeness and selectivity; (2) more complete and reliable methods have a greater impact on LLM behavior; and (3) nonlinear interventions are almost always more reliable than linear interventions.

Our project webpage is available at: https://ahdavies6.github.io/causal_probing_reliability/

1 Introduction

What latent properties do large language models (LLMs) learn to represent, and how do they leverage such representations? Causal probing aims to answer this question by intervening on a model's embedding representations of some property of interest (e.g., parts-of-speech), feeding the altered embeddings back into the LLM, and assessing how the model's behavior on downstream tasks changes (Geiger et al., 2020; Ravfogel et al., 2020; Elazar et al., 2021; Tucker et al., 2021; Lasri et al., 2022; Davies et al., 2023; Zou et al., 2023). However, it is only possible to draw meaningful conclusions about the model's use of the latent property if we are confident that interventions have fully and precisely carried out the intended transformation (Davies and Khakzar, 2024). Indeed, prior works have raised serious doubts about causal probing, finding that many intervention methods may have a large unintended impact on non-targeted properties (Kumar et al., 2022), and that the original value of the property may still be recoverable (Elazar et al., 2021; Ravfogel et al., 2022b). So far, it has been unclear how these doubts generalize to other types of interventions or how serious they are in practice, as there is no generally accepted approach for evaluating or comparing different methods.

Thus, our main goal in this study is to work toward a systematic understanding of the effectiveness and limitations of current causal probing methodologies. Specifically, we propose an empirical analysis framework to evaluate the *reliability* of causal probing according to two key desiderata:

1. *Completeness*: interventions should fully transform the representation of targeted properties.
2. *Selectivity*: interventions should not impact non-targeted properties.

We define completeness and selectivity using "validation probes" that enable measuring the impact of an intervention on both targeted and non-targeted properties. We apply our framework to several intervention methods and LLMs, observing that each method exhibits a clear tradeoff between these criteria. We also show that the most complete and reliable interventions lead to the largest and most consistent impact in LLM task performance. Finally, we find a substantial difference between the reliability of linear versus nonlinear interventions, where nonlinear methods are almost universally

*These authors contributed equally to this work.

Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pages 857–878

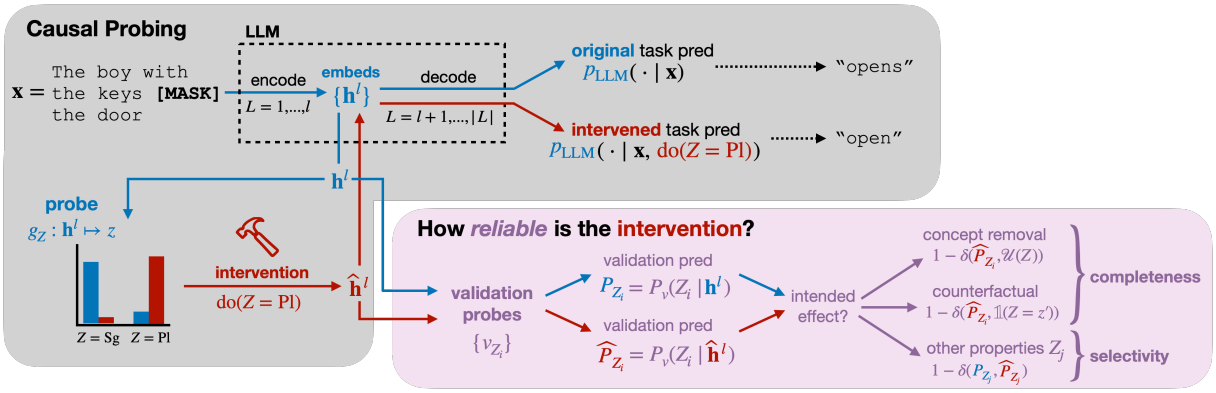


Figure 1: **Causal Probing and Our Reliability Framework.** The process of causal probing is shown in the gray box, with our reliability framework in the purple box.

- *Causal Probing:* embeddings $\mathbf{h}^l$ are extracted from layer $L = l$ of a model and used to train a probe g_Z to predict the value $Z = z$ of property Z from embeddings (e.g., the number of the subject, `boy`, is $Z = \texttt{Sg}$ for singular). A causal probing intervention $\text{do}(Z = \texttt{Pl})$ uses the probe g_Z to modify the representation encoded by $\mathbf{h}^l$ to encode plural instead. The resulting intervened embedding $\hat{\mathbf{h}}^l$ is fed back into the model at layer $L = l + 1$ and the forward pass is completed, changing the original prediction `opens` to the intervened prediction `open`.
- *Reliability Framework:* instead of feeding the intervened embedding $\hat{\mathbf{h}}^l$ back into the model, it is passed alongside $\mathbf{h}^l$ to validation probes $\{v_{Z_i}\}$ that independently test whether the intervention has had the intended effect. Completeness is measured as the similarity between the validation probe prediction and the target distribution for the intervention (e.g., a perfectly *complete* counterfactual intervention $\text{do}(Z = \texttt{Pl})$ would lead validation probe v_Z to predict plural with probability $P_v(Z = \texttt{Pl}|\hat{\mathbf{h}}^l) = 1$), and selectivity is the similarity between the validation probe distribution for non-targeted properties before and after the intervention (which, for a perfectly *selective* intervention, should not change).

more reliable than linear methods across LLMs and between different layers. This suggests that interventions relying on the linear representation hypothesis (see, e.g., Vargas and Cotterell, 2020; Ravfogel et al., 2020, 2022a; Tigges et al., 2023; Burns et al., 2023; Jiang et al., 2024; Park et al., 2024b,a) may yield inaccurate interpretations of model internals and behaviors. Finally, our framework also provides the first concrete basis for calibrating intervention hyperparameters to balance completeness and selectivity, allowing for more reliable interpretation of LLMs using existing methods.

2 Background and Related Work

Probing *Probing* aims to analyze which properties (e.g., part-of-speech, sentiment labels, etc.) are represented by a deep learning model (e.g., LLM) by training classifiers to predict these properties from latent embeddings (Belinkov, 2022). Given, say, an LLM M, input token sequence $\mathbf{x} = (x_1, ..., x_N)$, and embeddings $\mathbf{h}^l = M_l(\mathbf{x})$ of input $\mathbf{x}$ at layer l of M, suppose Z is a latent property of interest that takes a discrete value $Z = z$ for input $\mathbf{x}$. Here, the formal goal of probing is to train a classifier $g_Z : M_l(\mathbf{x}) \mapsto z$ to predict the value of Z from $\mathbf{h}^l$. On the most straightforward interpretation, if g_Z achieves high accuracy on the probe task, then the model is said to be "representing" Z. An important criticism of such claims is that *correlation does not imply causation* – i.e., that simply because a given property can be predicted from embedding representations does not mean that the model is using the property in any way (Hewitt and Liang, 2019; Elazar et al., 2021; Belinkov, 2022; Davies et al., 2023).

Causal Probing A prominent response to this concern has been *causal probing*, which uses probes to remove or alter that property in the model's representation, and measuring the impact of such interventions on the model's predictions (Elazar et al., 2021; Tucker et al., 2021; Lasri et al., 2022; Davies et al., 2023; see Figure 1). Specifically, causal probing performs interventions $\text{do}(Z)$ that modify M's representation of Z in the embeddings $\mathbf{h}^l$, producing $\hat{\mathbf{h}}^l$, where interventions can

either encode a counterfactual value $Z = z'$ (denoted $\text{do}(Z = z')$ where $z \neq z'$), or remove the representation of Z entirely (denoted $\text{do}(Z = 0)$). Following the intervention, modified embeddings $\hat{\mathbf{h}}^l$ are fed back into M beginning at layer $l + 1$ to complete the forward pass, yielding intervened predictions $\text{P}_M(\cdot|\mathbf{x}, \text{do}(Z))$. Comparison with the original predictions $\text{P}_M(\cdot|\mathbf{x})$ allows one to measure the extent to which M uses its representation of Z in computing them.

Causal Probing: Limitations Prior works have indicated that information about the target property that should have been completely removed may still be recoverable by the model (Elazar et al., 2021; Ravfogel et al., 2022b, 2023), in which case interventions are not complete; or that most of the impact of interventions may actually be the result of collateral damage to correlated, non-targeted properties (Kumar et al., 2022), in which case interventions are not selective. How seriously should we take such critiques? We observe several important shortcomings in each of these prior studies on the limitations of causal probing interventions:

1. These limitations have only been empirically demonstrated for the task of removing information about a target property from embeddings such that the model *cannot be fine-tuned to use the property for downstream tasks* (Kumar et al., 2022; Ravfogel et al., 2022b, 2023). But considering that the goal of causal probing is to interpret the behavior of an existing pre-trained model, the question is not whether models *can* be fine-tuned to use the property; it is whether models *already* use the property without task-specific fine-tuning, which has not been addressed in prior work. Do we observe the same limitations in this context?
2. These limitations have only been studied for linear concept removal interventions (e.g., Ravfogel et al. 2020, 2022a), despite the recent proliferation of other causal probing methodologies, including nonlinear (Tucker et al., 2021; Ravfogel et al., 2022b; Shao et al., 2022; Davies et al., 2023) and counterfactual interventions (Ravfogel et al., 2021; Tucker et al., 2021; Davies et al., 2023) (see Section 4). Do we observe the same limitations for, e.g., nonlinear counterfactual interventions?

In this work, we answer both questions by providing a precise, quantifiable, and sufficiently general definition of completeness and selectivity that it is applicable to *all* such causal probing interventions, and carry out extensive experiments to evaluate representative methods from each category of interventions when applied to a pre-trained LLM as it performs a zero-shot prompt task.

Causal Probing: Evaluation Note that, while we are the first to define and measure the completeness and selectivity of *causal probing interventions*, RAVEL (Huang et al., 2024) provides a broadly analogous evaluation framework and dataset with respect to *interchange interventions*. Methods for performing interchange interventions over embedding representations of a given property are trained on counterfactual minimal pairs of the property (i.e., two inputs which are identical in all respects except the input property; Geiger et al., 2020; Vig et al., 2020; Geiger et al., 2024). In contrast, causal probing, as studied in this work, does not require minimal pairs for training probes or performing interventions (Davies et al., 2023), allowing our empirical analysis to be carried out without access to such data.

3 Evaluating Causal Probing Reliability

Recall that our main goal in this work is to evaluate intervention reliability in terms of completeness (completely transforming M's representation of some target property Z_i) and selectivity (minimally impacting M's representation of other properties $Z_j \neq Z_i$).[1] Given that we cannot directly inspect what value M encodes for any given property Z_i, it is necessary to introduce the notion of *validation probes*, which we use to measure the extent to which interventions have fulfilled either criterion. Our complete reliability framework is visualized in Figure 1.

Validation Probes We define a validation probe v as a probe (trained independently from interventional probes; see Section 4) that returns a distribution $\text{P}_v(Z|\mathbf{h})$ over the values of property Z, and we interpret $\text{P}_v(Z = z|\mathbf{h})$ as the degree to which the model's embedding representations[2] $\mathbf{h}$ given

[1] In this paper, we use selectivity in the sense described by Elazar et al. (2021), and not other probing work such as Hewitt and Liang (2019), where it instead refers to the gap in performance between probes trained to predict real properties versus nonsense properties.

[2] For simplicity, we omit the superscript l denoting the layer embeddings $\mathbf{h}^l$ from which $\mathbf{h}$ is extracted; but our framework can be applied to study interventions over embeddings from any layer.

natural-language input $\mathbf{x}$ encodes a belief that $\mathbf{x}$ has the property $Z = z$. So, if $\mathbf{h}$ encodes value $Z = \hat{z}$ with complete certainty, v should return a degenerate distribution $\mathrm{P}_v(Z|\mathbf{h}) = \mathbb{1}(Z = \hat{z})$, whereas we would expect a uniform distribution $\mathrm{P}_v(Z|\mathbf{h}) = \mathcal{U}(Z)$ if $\mathbf{h}$ does not encode property Z at all.[3] Thus, validation probes enable us to estimate how well various intervention methods carry out the target transformation. (See Section 4 for details on validation probe training.)

Completeness If a counterfactual intervention $\mathrm{do}(Z = z')$ is perfectly *complete*, then it would produce a perfectly-intervened $\mathbf{h}^*_{Z=z'}$ that fully transforms $\mathbf{h}$ from encoding value $Z = z$ to encoding counterfactual value $Z = z' \neq z$. Thus, after performing the intervention, validation probe v should emit $\mathrm{P}_v(Z = z'|\mathbf{h}^*_{Z=z'}) = P^*_Z(Z = z') = 1$. For concept removal interventions $\mathrm{do}(Z = 0)$, a perfectly complete representation $\mathbf{h}^*_{Z=0}$ should not encode Z at all: $\mathrm{P}_v(Z|\mathbf{h}^*_{Z=0}) = P^*_Z(Z) = \mathcal{U}(Z)$.[4]

We can use any distributional distance metric $\delta(\cdot, \cdot)$ bounded by [0, 1] to determine how far the observed distribution $\hat{P}_Z = \mathrm{P}_v(Z|\hat{\mathbf{h}}_Z)$ is from the "goal" distribution P^*_Z. Throughout this work, we use total variation (TV) distance, which allows us to directly compare counterfactual and concept removal distributions: in both cases, $0 \leq c(\hat{\mathbf{h}}_Z) \leq 1$, where attaining 1 means the intervention had its intended effect in transforming the encoding of Z. Finally, for a given set of test embeddings $\mathbf{H} = \{\mathbf{h}^k\}^n_{k=1}$, the aggregate completeness over this test set $C(\mathbf{H}_Z)$ is the average $c(\hat{\mathbf{h}}^i_Z)$ across all $\mathbf{h}^k \in \mathbf{H}$.

For **counterfactual interventions**, we measure completeness as:

$$c(\hat{\mathbf{h}}_Z) = 1 - \delta(\hat{P}_Z, P^*_Z) \tag{1}$$

If the intervention is perfectly complete, then $\hat{P}_Z = P^*_Z$ and $c(\hat{\mathbf{h}}_Z) = 1$. On the other hand, if $\hat{P}_Z$ is maximally different from the goal distribution P^*_Z (e.g., $\hat{P}_Z = \mathrm{P}_v(Z = z|\hat{\mathbf{h}}_{Z=z'}) = 1$), then $c(\hat{\mathbf{h}}_Z) = 0$. For properties with more than two possible values, completeness is computed by averaging over each possible counterfactual value $z'_1, ..., z'_k \neq z$, yielding $c(\hat{\mathbf{h}}_Z) = \frac{1}{k}\sum_{i=1}^{k} \hat{c}(\mathbf{h}_{Z=z'_i})$.

For **concept removal interventions**, we measure completeness as:

$$c(\hat{\mathbf{h}}_Z) = 1 - \frac{k}{k-1} \cdot \delta(\hat{P}_Z, P^*_Z) \tag{2}$$

where k is the number of values Z can take. The normalizing factor is needed because P^*_Z is the uniform distribution over k values and hence $0 \leq \delta(\hat{P}_Z, P^*_Z) \leq 1 - \frac{1}{k}$.

Selectivity If an intervention on property Z_i is *selective*, the intervention should not impact M's representation of any non-targeted property $Z_j \neq Z_i$. Thus, for both counterfactual and concept removal interventions, validation probe v's prediction for any such Z_j should not change after the intervention.

To measure the selectivity of a modified representation $\hat{\mathbf{h}}_{Z_i}$ with respect to Z_j, denoted $s_j(\hat{\mathbf{h}}_{Z_i})$, we can again measure the distance between the observed distribution $\hat{P}_{Z_j} = \mathrm{P}_v(Z_j|\hat{\mathbf{h}}_{Z_i})$ and the original (non-intervened) distribution $P_{Z_j} = \mathrm{P}_v(Z_j|\mathbf{h})$:

$$s_j\big(\hat{\mathbf{h}}_{Z_i}\big) = 1 - \frac{1}{m} \cdot \delta\big(\hat{P}_{Z_j}, P_{Z_j}\big) \tag{3}$$

$$\text{where } m = \max\big(1 - \min(P_{Z_j}), \max(P_{Z_j})\big)$$

Since $0 \leq \delta(\hat{P}_{Z_j}, P_{Z_j}) \leq m$, we divide by m to normalize selectivity to $0 \leq s_j(\hat{\mathbf{h}}_{Z_i}) \leq 1$. If multiple non-targeted properties $Z_{j_1}, ..., Z_{j_{\text{max}}}$ are being considered, selectivity $s(\hat{\mathbf{h}}_{Z_i})$ is computed as the average over all such properties $s_{j_m}(\hat{\mathbf{h}}_{Z_i})$. Finally, analogous to completeness, the aggregate selectivity over a set of test embeddings $\mathbf{H}_{Z_i} = \{\mathbf{h}^k\}^n_{k=1}$, denoted $S(\mathbf{H}_{Z_i})$, is the average selectivity $s(\hat{\mathbf{h}}^k_{Z_i})$ across all $\mathbf{h}^k_{Z_i} \in \mathbf{H}_{Z_i}$.

Reliability Since completeness and selectivity can be seen as a trade-off, we define the overall reliability of an intervention $R(\hat{\mathbf{H}})$ as the harmonic mean of $C(\hat{\mathbf{H}}^l)$ and $S(\hat{\mathbf{H}}^l)$. This is analogous to the F1-score, which is the harmonic mean of precision and recall: just as a degenerate classifier can achieve perfect recall and low precision by always predicting the positive class, a degenerate intervention can achieve perfect selectivity and low completeness by performing no intervention at all.

[3] A validation probe's prediction is subtly different from the prediction an arbitrary classifier should make in the absence of any evidence about Z: such a classifier should revert to the empirical distribution $\hat{\mathrm{P}}(Z)$.

[4] This is only expected when using concept removal interventions for *causal probing* – i.e., when intervening on a model's representation and feeding it back into the model to observe how the intervention modifies its behavior. When considering concept removal interventions for *concept removal* (a more common setting), a more appropriate "goal" distribution P^*_Z would be $\mathrm{P}(Z)$, the label distribution. See Appendix A for further discussion.

Using harmonic mean to calculate reliability heavily penalizes such interventions.

4 Experimental Setting

LLMs We test our framework in experiments across six language models: BERT (Devlin et al., 2019), GPT-2 (Radford et al., 2019), three Pythia models (160M, 1.4B, and 6.9B; Biderman et al., 2023), and Llama 3.2 (3B, instruction-tuned; Grattafiori et al., 2024). We include BERT and GPT-2 to test causal probing methods in more traditional settings they were originally designed for – e.g., BERT (an encoder-only masked language model) has been very extensively studied in causal probing (Ravfogel et al., 2020; Rogers et al., 2021; Elazar et al., 2021; Ravfogel et al., 2021; Lasri et al., 2022; Ravfogel et al., 2022b, 2023; Davies et al., 2023), and many methods have been designed specifically with this model in mind. We include the range of Pythia models to study how these methods scale and generalize to the popular GPT-like family of architectures (decoder-only models trained on autoregressive language modeling). Finally, we account for the effect of popular post-training techniques like instruction-tuning and RLHF by studying the selected Llama model.[5]

Task Following several prior causal probing works (Lasri et al., 2022; Ravfogel et al., 2021; Arora et al., 2024), we select the prompting task of **subject-verb agreement**. (In Appendix C.5, we also repeat some experiments for the IOI task introduced by Wang et al. 2023.) In subject-verb agreement, each data point takes the form $\langle \mathbf{x}_i, y_i \rangle$ where $\mathbf{x}_i$ is a sentence such as "the boy with the keys `[MASK]` the door," and the task of the LLM is to predict $\mathrm{P}_M(y_i|\mathbf{x}) > \mathrm{P}_M(y_i'|\mathbf{x})$ (here, that $y_i =$ "locks" rather than $y_i' =$ "lock") – see the example in Figure 1. The causal variable Z_c is the number of the subject (`Sg` or `Pl`), because (grammatically) this is the only variable that determines the number of the verb in English. The environmental (non-causal) variable Z_e is the number (`Sg` or `Pl`) of the noun immediately preceding the verb to be conjugated when that noun is not the subject (e.g., "keys" in the phrase "with the keys"). Note that, in this work, we consider a task in the simplest experimental setting (two binary properties) that allows us to study interventions using our framework; however, nothing in our methodology precludes the use of more properties, or properties with more possible values.

Dataset We use the LGD dataset (Linzen et al., 2016), which consists of >1M naturalistic English sentences from Wikipedia; from this we take only sentences for which both singular and plural forms of the target verb are in LLMs' vocabularies. We use 40% of the examples to train validation probes, 40% to train interventional probes, and 20% as a test set. (More dataset details can be found in Appendix B.1.)

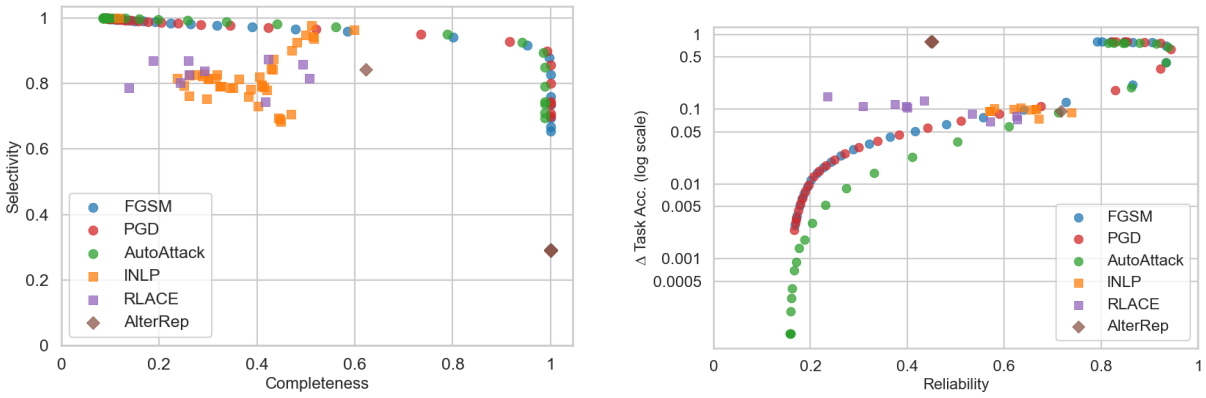
Validation Probes For each layer l and probed property Z, we experiment with several instantiations of validation probes, including linear and MLP probes across a range of hyperparameters (Appendix B.2), observing similar results between them (see Appendix C.3). Thus, for all results reported in the main paper, we default to the validation probe architecture and hyperparameters with the highest validation-set accuracy for the probed property.[6] Validation probes are trained on data that is completely disjoint from that used to train interventional probes, and where Z_c and Z_e are made independent by subsampling the largest (random) subset that preserves label distributions $\mathrm{P}(Z_c), \mathrm{P}(Z_e)$. This is important for validation probes to serve as unbiased arbiters of selectivity, as spurious correlations between the variables could lead a probe that is trained on property Z_c to partially rely on representations of Z_e (Kumar et al., 2022).[7]

Interventions We explore two (linear) concept removal interventions: INLP (Ravfogel et al., 2020), which iteratively trains classifiers on Z and projects embeddings into their nullspaces; and RLACE (Ravfogel et al., 2022a), which identifies a minimal-rank subspace to remove information that is linearly predictive of Z by solving a constrained minimax game. We explore one linear

[5] For information on post-training of the `Llama-3.2-3B-Instruct` model used in our experiments, see: https://huggingface.co/meta-llama/Llama-3.2-3B-Instruct#training-data

[6] Note that this always results in MLP validation probes; see Appendix C.3 for results with linear validation probes. Validation sets used for selecting validation probes have the same independence and label-distribution properties as their train sets.

[7] We leave these spurious correlations in training data of interventional probes to test the impact that they have on the resulting interventions' completeness and selectivity, as this is a better proxy for their completeness and selectivity in more realistic settings where controlling for spurious correlations may not always be possible.



(a) **Selectivity vs. Completeness** (b) **ΔTask Acc vs. Reliability**

Figure 2: **Completeness, selectivity, reliability, and ΔTask Acc** for all interventions in the **final layer of Pythia-160M.** Each point in both plots corresponds to a different hyperparameter setting. (Appendix C.1 contains analogous results for all other models.)

counterfactual method, AlterRep (Ravfogel et al., 2021), which builds on INLP by projecting embeddings along classifiers' rowspaces, placing them on the counterfactual side of the separating hyperplanes. Finally, we study three nonlinear counterfactual methods, which are all gradient-based interventions (Davies et al., 2023): a MLP probe is trained on Z, then gradient-based ("white-box") adversarial attacks are applied to minimize the loss of the probe with respect to the target counterfactual value $Z = z'$ within an L_∞-ball of radius ε around the original embedding. We experiment with three gradient attack methods – FGSM (Goodfellow et al., 2015), PGD (Madry et al., 2017), and AutoAttack (Croce and Hein, 2020) – as described in Appendix B.3. After intervening on Z_c to obtain representations $\hat{\mathbf{h}}_{Z_c}$, we use validation probes $\hat{o}$ to measure completeness, selectivity, and reliability. Due to compute limitations, we restrict our analysis for Pythia-1.4B and -6.9B to three of the six methods: INLP, AlterRep, and FGSM. Note that this includes at least one method from each of the three classes of methods defined above (linear removal, linear counterfactual, and nonlinear counterfactual, respectively).

Impact on Model Behavior The ultimate goal of causal probing is to measure a model M's use of a property Z by comparing intervened predictions $\mathrm{P}_M(\cdot|\mathbf{x}, \mathrm{do}(Z))$ to its original predictions $\mathrm{P}_M(\cdot|\mathbf{x})$. Our framework aims to measure the reliability of the interventions themselves, a prerequisite to making claims about the underlying model. It is nonetheless important to consider how the completeness, selectivity, and reliability of a given intervention relate to its impact on model behavior. Thus, for each intervention, we also feed intervened final-layer embeddings $\hat{\mathbf{h}}^L$ for all test instances back into models immediately before word prediction, measuring task accuracy based on whether they assign the correct verb form a higher probability, and subtract this "intervened" accuracy from the original task accuracy (98.62%) for each intervention to yield ΔTask Acc (cf. Elazar et al., 2021; Lasri et al., 2022; Davies et al., 2023).

5 Experimental Results

Below, we present results for completeness, selectivity, reliability, and ΔTask Acc of all intervention methods in models' final layer (Section 5.1), then examine their reliability in earlier layers (Section 5.2). Note that, while we only have space to include plots for Pythia-160M (henceforth referred to as "Pythia") in this section of the main paper, analogous plots for the other models are available in Appendix C.

5.1 Final-Layer Results

First, we note that both validation probes are able to consistently predict each property (97.3% and 94.4% accuracy for Z_c and Z_e, respectively), which is a necessary prerequisite to validate any further results.

Completeness, Selectivity, & Reliability Each intervention has a hyperparameter (ε for GBIs, rank r for INLP and RLACE, and α for AlterRep), where increasing its value leads to stronger interventions. Thus, each hyperparameter setting yields a different value of completeness, selectivity, and

	BERT	GPT2	Pythia-160M	Pythia-1.4B	Pythia-6.9B	Llama-3.2-3B
INLP	0.464	0.106	0.739	0.841	0.751	0.668
RLACE	0.429	0.495	0.627			
AlterRep	**0.835**	0.427	0.717	0.597	0.519	0.891
FGSM	0.552	0.958	0.934	**0.920**	**0.951**	**0.960**
PGD	0.509	**0.98**	**0.943**			
AutoAttack	0.514	0.97	0.938			

Table 1: **Intervention scores for maximum-reliability hyperparameters** in the **final layer** of all models. (See Section 5.2 for results in earlier layers.) Scores are reported for the hyperparameter x_{opt} that maximizes the reliability of each respective method, and the highest-reliability method for each model is bolded.

reliability for a given intervention. Figure 2a plots selectivity against completeness for each method in Pythia's final layer, showing that increasing the hyperparameter values yields higher completeness and lower selectivity. (Analogous results for other models, as well as plots of completeness, selectivity, and reliability broken down by method and hyperparameter value, are available in Appendix C.1.)

Table 1 shows these metrics for each method at the hyperparameter that yields the highest reliability. For all models other than BERT, the nonlinear counterfactual methods (GBIs: FGSM, PGD, and AutoAttack) have the highest overall reliability; for BERT only, AlterRep is most reliable; and otherwise, the linear methods (both removal and counterfactual) tend to show middling reliability, varying from a low of 0.106 for INLP on GPT2 to a high of 0.841 for INLP on Pythia-1.4B.

Task Accuracy Figure 2b shows ΔTask Acc as a function of the reliability for each intervention and hyperparameter setting. For most methods and hyperparameter values, ΔTask Acc increases alongside intervention reliability. Notably, the points at which the GBIs (FGSM, PGD, and AutoAttack) achieve the highest ΔTask Acc are *not* at their highest reliability values, resulting in a backward curve visible at the top of Figure 2b, corresponding to hyperparameter ε being raised past the point of maximum reliability where there is near-perfect completeness but much lower selectivity (see Appendix C.1). Finally, RLACE and INLP shows a similar impact on task accuracy even for different completeness and reliability scores due to its "noisy" equilibrium in reliability and completeness for high rank r (see Appendix C.1).

5.2 Reliability by Layer

As in the final layer, validation probes over earlier layers can consistently predict each property with high accuracy (see Appendix C.4). Figure 3 shows the reliability for each method using the hyperparameters that obtain the highest reliability in that layer. Across all layers, each nonlinear counterfactual method (GBIs: FGSM, PGD, and AutoAttack) is more reliable than all linear methods. We also observe this trend for all other models Appendix C.2 (with the exception of BERT, for which AlterRep is more reliable than the GBIs in layers 10-12; see Figure 9).

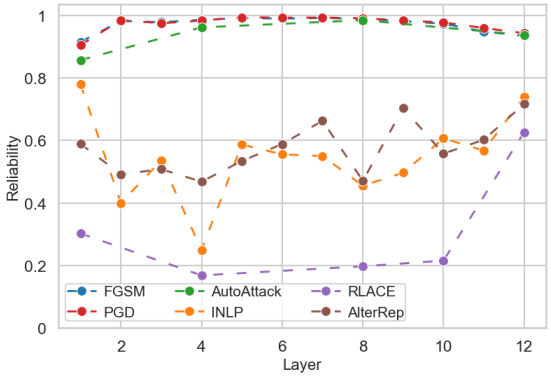


Figure 3: **Maximum reliability by layer** for each intervention across **all layers of Pythia-160M.** (Appendix C.2 contains analogous results for all other models.)

6 Discussion

Tradeoff: Completeness vs. Selectivity In Pythia's final layer, no method is able to achieve perfect completeness without sacrificing selectivity (see Figure 2a), a trend which we also see for all other models (see Appendix C.1). However, we observe a very favorable tradeoff for GBIs, which incur only a small selectivity cost for increasing completeness, leading to high overall reliability across all models and layers (see Appendix C.2). In contrast, linear removal methods (INLP and RLACE) tend to have much higher selectivity than they do

completeness, which is likely because these methods are explicitly optimized to minimize collateral damage (Ravfogel et al., 2020, 2022a), despite being linear and potentially incomplete (with respect to putatively nonlinear representations). Finally, the linear counterfactual method (AlterRep) tends to have highly variable behavior between different models: for Pythia-160M, Llama 3.2, and GPT2, it shows middling performance across layers; it is the *most* reliable method in BERT's last 3 layers, with high selectivity and near-perfect completeness; and it is the *least* reliable method in the later layers of Pythia-1.4B and Pythia-6.9B.

Reliability and Task Accuracy Figures 2b, 4b and 5b show a clear trend: more complete interventions and hyperparameter values show a greater impact on task performance. In particular, counterfactual methods (GBIs and AlterRep), which show consistently higher completeness than removal methods (INLP and RLACE), also show (near-)total ΔTask Acc. This is highly intuitive: in the case where models perform the subject-verb agreement task by leveraging its representation of Z_c, then more complete interventions would have a greater effect on the model's task performance. We do not claim that this is necessarily the case – e.g., our results might have looked different if we had intervened in earlier layers; and the primary object of our study is the completeness, selectivity, and reliability of the causal probing methods we have experimented with, not the representations used by LLMs to perform a simple grammatical task. Rather, we take the clear relationship between intervention completeness and ΔTask Acc to be a strong indicator that more complete methods indeed yield stronger results, reinforcing the utility of our framework in evaluating causal probing interventions as tools for studying models' use of latent representations. In particular, our framework provides the first concrete approach for calibrating intervention hyperparameters in the latent space (i.e., max-reliability hyperparameter search using validation probes), allowing researchers to adaptively balance the priorities of completeness and selectivity and examine the corresponding effect on model behaviors, rather than simply resorting to maximum-strength (Tucker et al., 2021) or minimum-collateral damage (Ravfogel et al., 2020, 2022a) interventions.

Linearity by Layer Overall, the nonlinear GBI methods are more reliable than the linear methods across all models and layers, (with the sole exception of BERT in layers 10-12; see Appendix C.2). Without adequate controls, this might simply be the result of using MLP validation probes in all results reported in the main paper, which could bias our analysis in favor of nonlinear methods, as validation probes and nonlinear interventional probes might be relying on similarly-encoded information and neglecting linearly-encoded information. We account for this possibility by repeating layerwise reliability experiments using linear validation probes in Appendix C.3, finding that they show remarkably similar results to MLP validation probes.

Thus, we briefly consider the more interesting possibility that the reliability gap between linear and nonlinear LLMs *may be due to LLMs encoding task-relevant representations nonlinearly*, particularly in intermediate layers: for instance, in addition to the aforementioned example of BERT's last 3 layers, Pythia-160M also shows that all linear methods are substantially more reliable in the first and last layers than they are in intermediate layers. While this conjecture is not fully supported by all results (e.g., INLP and AlterRep drop substantially in reliability in GPT2's final layer), it is nonetheless intuitive that some models may be more nonlinear in intermediate layers than their final layer, as embeddings in earlier layers will pass through many nonlinearities before word prediction, allowing a high degree of nonlinear representation (White et al., 2021); whereas any output-discriminative information must be made linearly separable in the final embedding layer of neural networks (Alain and Bengio, 2017). There is a long history of work studying the so-called *linear representation hypothesis* (LRH; Mikolov et al., 2013; Pennington et al., 2014; Bolukbasi et al., 2016; Vargas and Cotterell, 2020) – i.e., that neural networks encode most or all features linearly – with some recent works suggesting that this hypothesis is true even for modern LLMs (Burns et al., 2023; Tigges et al., 2023; Park et al., 2024b). However, many of these studies often consider embeddings only in the input or final ("unembedding") layer of LLMs (Jiang et al., 2024; Park et al., 2024b,a), neglecting intermediate layers. Our findings provide an important contrast: while they do not directly validate or refute the LRH, the stark difference between the reliability of linear and nonlinear counterfactual methods indicates that it is critical to consider multiple layers throughout models when studying the

LRH, as findings of linearity in the final layer may not generalize to earlier layers.

7 Conclusion

In this work, we proposed a general empirical evaluation framework for causal probing, defining the reliability of interventions in terms of completeness, selectivity, and reliability. Our framework makes it possible to directly compare different kinds of interventions, such as linear vs. nonlinear or counterfactual vs concept removal methods. We applied our framework to study leading causal probing techniques across a range of LLMs, finding that they all exhibit a tradeoff between completeness and selectivity, that more reliable and complete methods yield a greater impact on LLM task performance, and that nonlinear methods tend to be much more reliable than linear methods. Finally, we explored the implications of these findings for future work in optimizing intervention hyperparameters and studying the linear representation hypothesis.

Limitations

An important empirical limitation of our work is that we only study the relatively simple subject-verb agreement task (and IOI; see Appendix C.5). We intentionally select simple, well-studied syntactic tasks with a single binary causal variable and one binary environmental variable, opting for a more parsimonious task in this setting to avoid introducing exogenous confounds while studying a novel latent-space evaluation framework in the context of several highly distinct families of methods. Selecting simple tasks also allows for easy comparison between a range of LLMs at different scales (which are all able to solve the task nearly perfectly). However, now that we have validated our framework in the context of these simple tasks, it will be important to extend this study to more complex and interesting tasks, such as those with multiple causal variables that take an arbitrary number of possible values, or those that even frontier models struggle to solve (for use in "debugging" what representations are being learned and used by LLMs in performing difficult tasks). We aim to explore such settings in future work.

Acknowledgements

This research used the Delta advanced computing and data resource which is supported by the National Science Foundation (award OAC 2005572) and the State of Illinois. Delta is a joint effort of the University of Illinois Urbana-Champaign and its National Center for Supercomputing Applications. This work also utilizes resources supported by the National Science Foundation's Major Research Instrumentation program, grant #1725729, as well as the University of Illinois at Urbana-Champaign. Adam Davies is supported by the National Science Foundation and the Institute of Education Sciences, U.S. Department of Education, through Award #2229612 (National AI Institute for Inclusive Intelligent Technologies for Education). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of National Science Foundation or the U.S. Department of Education.

References

Guillaume Alain and Yoshua Bengio. 2017. Understanding intermediate layers using linear classifier probes.

Aryaman Arora, Dan Jurafsky, and Christopher Potts. 2024. Causalgym: Benchmarking causal interpretability methods on linguistic tasks. *arXiv preprint arXiv:2402.12560*.

Yonatan Belinkov. 2022. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, and 1 others. 2023. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, pages 2397–2430. PMLR.

Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. 2016. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. *Advances in neural information processing systems*, 29.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. 2023. Discovering latent knowledge in language models without supervision. In *The Eleventh International Conference on Learning Representations*.

Francesco Croce and Matthias Hein. 2020. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International conference on machine learning*, pages 2206–2216. PMLR.

Adam Davies, Jize Jiang, and ChengXiang Zhai. 2023. Competence-based analysis of language models. *arXiv preprint arXiv:2303.00333*.

Adam Davies and Ashkan Khakzar. 2024. The cognitive revolution in interpretability: From explaining behavior to interpreting representations and algorithms. *arXiv preprint arXiv:2408.05859*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Yanai Elazar, Shauli Ravfogel, Alon Jacovi, and Yoav Goldberg. 2021. Amnesic probing: Behavioral explanation with amnesic counterfactuals. *Transactions of the Association for Computational Linguistics*, 9:160–175.

Atticus Geiger, Kyle Richardson, and Christopher Potts. 2020. Neural natural language inference models partially embed theories of lexical entailment and negation. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 163–173, Online. Association for Computational Linguistics.

Atticus Geiger, Zhengxuan Wu, Christopher Potts, Thomas Icard, and Noah Goodman. 2024. Finding alignments between interpretable causal variables and distributed neural representations. In *Proceedings of the Third Conference on Causal Learning and Reasoning*, volume 236 of *Proceedings of Machine Learning Research*, pages 160–187. PMLR.

Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

John Hewitt and Percy Liang. 2019. Designing and interpreting probes with control tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2733–2743, Hong Kong, China. Association for Computational Linguistics.

Jing Huang, Zhengxuan Wu, Christopher Potts, Mor Geva, and Atticus Geiger. 2024. RAVEL: Evaluating interpretability methods on disentangling language model representations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8669–8687, Bangkok, Thailand. Association for Computational Linguistics.

Yibo Jiang, Goutham Rajendran, Pradeep Kumar Ravikumar, Bryon Aragam, and Victor Veitch. 2024. On the origins of linear representations in large language models. In *Forty-first International Conference on Machine Learning*.

Abhinav Kumar, Chenhao Tan, and Amit Sharma. 2022. Probing classifiers are unreliable for concept removal and detection. *Advances in Neural Information Processing Systems*, 35:17994–18008.

Karim Lasri, Tiago Pimentel, Alessandro Lenci, Thierry Poibeau, and Ryan Cotterell. 2022. Probing for the usage of grammatical number. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8818–8831, Dublin, Ireland. Association for Computational Linguistics.

Tal Linzen, Emmanuel Dupoux, and Yoav Goldberg. 2016. Assessing the Ability of LSTMs to Learn Syntax-Sensitive Dependencies. *Transactions of the Association for Computational Linguistics*, 4:521–535.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2017. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.

Tomáš Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: Human language technologies*, pages 746–751.

Kiho Park, Yo Joong Choe, Yibo Jiang, and Victor Veitch. 2024a. The geometry of categorical and hierarchical concepts in large language models. *arXiv preprint arXiv:2406.01506*.

Kiho Park, Yo Joong Choe, and Victor Veitch. 2024b. The linear representation hypothesis and the geometry of large language models. In *Forty-first International Conference on Machine Learning*.

A Paszke. 2019. Pytorch: An imperative style, high-performance deep learning library. *arXiv preprint arXiv:1912.01703*.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, and 1 others. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. 2020. Null it out: Guarding protected attributes by iterative nullspace projection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7237–7256, Online. Association for Computational Linguistics.

Shauli Ravfogel, Yoav Goldberg, and Ryan Cotterell. 2023. Log-linear guardedness and its implications. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9413–9431, Toronto, Canada. Association for Computational Linguistics.

Shauli Ravfogel, Grusha Prasad, Tal Linzen, and Yoav Goldberg. 2021. Counterfactual interventions reveal the causal effect of relative clause representations on agreement prediction. In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 194–209, Online. Association for Computational Linguistics.

Shauli Ravfogel, Michael Twiton, Yoav Goldberg, and Ryan D Cotterell. 2022a. Linear adversarial concept erasure. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 18400–18421. PMLR.

Shauli Ravfogel, Francisco Vargas, Yoav Goldberg, and Ryan Cotterell. 2022b. Adversarial concept erasure in kernel space. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6034–6055, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2021. A primer in bertology: What we know about how bert works. *Transactions of the Association for Computational Linguistics*, 8:842–866.

Shun Shao, Yftah Ziser, and Shay B. Cohen. 2022. Gold doesn't always glitter: Spectral removal of linear and nonlinear guarded attribute information. *arXiv preprint*.

Curt Tigges, Oskar John Hollinsworth, Atticus Geiger, and Neel Nanda. 2023. Linear representations of sentiment in large language models. *arXiv preprint arXiv:2310.15154*.

Mycal Tucker, Peng Qian, and Roger Levy. 2021. What if this modified that? syntactic interventions with counterfactual embeddings. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 862–875, Online. Association for Computational Linguistics.

Francisco Vargas and Ryan Cotterell. 2020. Exploring the linear subspace hypothesis in gender bias mitigation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2902–2913, Online. Association for Computational Linguistics.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Simas Sakenis, Jason Huang, Yaron Singer, and Stuart Shieber. 2020. Causal mediation analysis for interpreting neural nlp: The case of gender bias. *arXiv preprint arXiv:2004.12265*.

Kevin Ro Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. 2023. Interpretability in the wild: a circuit for indirect object identification in GPT-2 small. In *The Eleventh International Conference on Learning Representations*.

Jennifer C. White, Tiago Pimentel, Naomi Saphra, and Ryan Cotterell. 2021. A non-linear structural probe. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 132–138, Online. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and 1 others. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Fred Zhang and Neel Nanda. 2024. Towards best practices of activation patching in language models: Metrics and methods. In *The Twelfth International Conference on Learning Representations*.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, and 1 others. 2023. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*.

A Framework Details

Completeness of Concept Removal Interventions In Equation (2), we define the "goal" distribution P^*_Z of a concept removal intervention used in causal probing as being the uniform distribution – i.e., for a perfect concept removal intervention, $P^*_Z = \mathrm{P}_v(Z|\mathbf{h}^*_{Z=0}) = \mathcal{U}(Z)$. However, this is only true in the case of *causal probing*, and is not true of some concept removal applications such as guarding protected attributes (see, e.g., Ravfogel et al. 2023). That is, in the case of causal probing, the goal of an intervention is to intervene on a model's representation during its forward pass, feeding the intervened embedding back into the model and observing the change in the model's behavior (as described in Section 2). Recall that the purpose of a validation probe v is to decode model M's representation of a given property Z, not to predict its ground truth value – that is, even if M encodes the incorrect value of $Z = z'$ rather than $Z = z$ for a given input, the validation probe should still decode the incorrect value $Z = z'$. Indeed, this is precisely the principle behind using validation probes in the case of counterfactual interventions that change the representation of $Z = z$ to counterfactual value $Z = z'$, where validation probes are used to validate the extent to which the representation has actually been changed to encode this counterfactual value, and the ideal counterfactual intervention yields $\mathrm{P}_v(Z = z'|\mathbf{h}^*_{Z=z'}) = P^*(Z = z') = 1$. However, in the case of concept removal interventions $\mathrm{do}(Z = 0)$, an intervened embedding $\mathbf{h}^*_{Z=0}$ would ideally remove all information encoding M's representation of the value taken by Z, meaning that the M would not encode any value $Z = z_1$ as being more probable than $Z = z_2$ (as any information that is predictive of the value taken by Z should have been removed). In this case, the validation probe v would predict an equal probability $\mathrm{P}_v(Z = z_i|\mathbf{h}^*_{Z=0})$ for any given value z_i that may be taken by Z_i – i.e., $\mathrm{P}_v(Z|\mathbf{h}^*_{Z=0}) = P^*_Z = \mathcal{U}(Z)$.

However, this is not the case in the context of instances such as guarding protected attributes, where the goal of an intervention $\mathrm{do}(Z = 0)$ is to remove all information that is predictive of Z from embedding representations $\mathbf{h}^*_{Z=0}$ *such that no probe g can be trained to predict* $\mathrm{P}_g(Z|\mathbf{h}^*_{Z=0})$ any better than predicting $\mathrm{P}(Z)$ – i.e., ignoring the embedding entirely and simply mapping every input to the label distribution $\mathrm{P}(Z)$ (Ravfogel et al., 2023). In this case, the probe g is trained on intervened embeddings $\hat{\mathbf{h}}_{Z=0}$, in which case it can learn to map every such embedding to the label distribution $\mathrm{P}(Z)$, which yields superior performance relative to predicting the uniform distribution $\mathcal{U}(Z)$ in any case where the label distribution $\mathrm{P}(Z)$ is not perfectly uniform, as such a g would have an expected accuracy equal to the proportion of test instances with the most common label $Z = z_{\text{argmax}}$ (which would be greater than the accuracy $\frac{1}{k}$ expected by defaulting to $\mathcal{U}(Z)$).

The key technical distinction between these two use cases of concept removal interventions is *whether or not probes or underlying models are trained or fine-tuned in the context of interventions*. In the case of causal probing, they are not – the (frozen) model M has no opportunity to recover the original value of $Z = z$ following a concept removal intervention $\mathrm{do}(Z = 0)$, and this should be reflected by validation probes. This is natural, given that the purpose of causal probing is to interpret the properties used by M in making a given prediction, not to test whether M can be trained to recover properties removed by interventions; and this is reflected by validation probes v, which are never trained on intervened embeddings. In contrast, for concept removal, probes (or models) are trained on intervened embeddings, and may learn to recover properties removed by interventions, meaning that – even in the worst case where all information has been removed – it would at least be possible to learn to reproduce the label distribution $\mathrm{P}(Z)$; but there is no reason to expect a model M or validation probe v to do so, given that they have never been trained on intervened embeddings. Thus, while we define the "goal" distribution $P^*_Z = \mathcal{U}(Z)$ for measuring the completeness of concept removal interventions as being $\mathcal{U}(Z)$ rather than $\mathrm{P}(Z)$, this distribution would instead be $P^*_Z = \mathrm{P}(Z)$ in the case of concept removal.

B Experimental Details

B.1 LGD Dataset

We use syntax annotations to extract values for the environmental variable Z_e from the LGD dataset (Linzen et al., 2016): if the part-of-speech of the word immediately preceding the `[MASK]` token is a noun,

	$Z_e = \emptyset$	$Z_e = \texttt{Sg}$	$Z_e = \texttt{Pl}$	Total
$Z_c = \texttt{Sg}$	176K	31K	5K	213K
$Z_c = \texttt{Pl}$	78K	10K	4K	92K
Total	254K	41K	9K	305K

Table 2: **Contingency Table on Test Set**. Distribution of data across combinations of causal and environmental variables. $Z_e = \emptyset$ denotes instances which have no prepositional phrase attached to the subject (and thus, contain no environmental variable). Note that the label distributions are unbalanced: $\mathrm{P}(Z_c = \texttt{Sg}) = 69.8\%$ and $\mathrm{P}(Z_e = \texttt{Sg} | E \neq \emptyset) = 81.5\%$.

and it is the object of a preposition (i.e., not the subject), then its number defines Z_e. About 83% of the sentences do not have a prepositional object preceding `[MASK]`, and so are only relevant for causal interventions.

The contingency table for values of Z_c and Z_e in the test set are in Table 2.

B.2 Probe Details

Our experiments include linear and MLP probes (both for interventions and as validation probes). Linear interventions (INLP, RLACE, and AlterRep) require linear probes; and for nonlinear interventions (GBIs), we use MLPs. We implement probes using PyTorch (Paszke, 2019), and leverage LLM implementations of all models available via HuggingFace Transformers (Wolf et al., 2019). For validation probes, we experiment with both linear and MLP probes. For all probes, we select hyperparameters by performing a grid search across candidate hyperparameter values, selecting the hyperparameters that yield the highest validation-set accuracy. We save probe parameters from the epoch with the highest validation-set accuracy with patience of 4 epochs. All probes are trained with cross-entropy loss.

For all **linear probes**, we consider learning rates in [0.0001, 0.001, 0.01, 0.1].

For **MLP probes**, we perform grid search over the following hyperparameter values:

- Number of hidden layers: [1, 2, 3]
- Layer size: [64, 256, 512, 1024]
- Learning rate: [0.0001, 0.001, 0.01]

B.3 Interventions

Gradient Based Interventions For all gradient-based intervention methods (Davies et al., 2023), we define the maximum perturbation magnitude of each intervention as ε (i.e., $||\hat{\mathbf{h}}_Z - \mathbf{h}||_\infty \leq \varepsilon$), and experiment over a range of ε values between 0.005 to 5.0 – specifically, $\varepsilon \in [0.005, 0.006, 0.007, 0.009, 0.011, 0.013, 0.016, 0.019, 0.024, 0.029, 0.035, 0.042, 0.051, 0.062, 0.076, 0.092, 0.112, 0.136, 0.165, 0.2, 0.286, 0.409, 0.585, 0.836, 1.196, 1.71, 2.445, 3.497, 5.0]$. We consider the following gradient attack methods for GBIs:

1. **FGSM** We implement Fast Gradient Sign Method (FGSM; Goodfellow et al., 2015) interventions as:

$$h' = h + \varepsilon \cdot \text{sgn}\left(\nabla_h \mathcal{L}\left(f_{\text{cls}}, x, y\right)\right)$$

2. **PGD** We implement Projected Gradient Descent (PGD; Madry et al., 2017) interventions as $h' = h^T$ where

$$h_{t+1} = \Pi_{\mathcal{N}(h)}\left(h_t + \alpha \cdot \text{sgn}\left(\nabla_h \mathcal{L}(f_{\text{cls}}, x, y)\right)\right)$$

   for iterations $t = 0, 1, \ldots, T$, projection operator Π, and L_∞-neighborhood $\mathcal{N}(h) = \{h' : \|h - h'\| \leq \varepsilon\}$. For PGD, we use 2 additional hyperparameters: iterations T and step size α, while fixing $T = 40$, as suggested by (Davies et al., 2023).
3. **AutoAttack** AutoAttack (Croce and Hein, 2020) is an ensemble of adversarial attacks that includes FAB, Square, and APGD attacks. Auto-PGD (APGD) is a variant of PGD that automatically adjusts the step size to ensure effective convergence. The parameters used were set as norm $= L_\infty$ and for Square attack, the n_queries=5000.

Concept Removal Interventions For concept removal interventions, we project embeddings into the nullspaces of classifiers. Here, the the rank r corresponds to the dimensionality of the subspace identified and erased by the intervention, meaning that the number of dimensions removed is equal to the rank.[8] We experiment over the range of values $r \in [0, 1, ..., 40]$. We consider the following concept removal interventions:

1. **INLP** We implement Iterative Nullspace Projection (INLP; Ravfogel et al., 2020) as follows: we train a series of classifiers $w_1, ..., w_n$, where in each iteration, embeddings are projected into the nullspace of the preceding classifiers $P_N(w_0) \cap \cdots \cap P_N(w_n)$. We then apply the combined projection matrix to calculate the final projection where $P := P_N(w_1) \cap \cdots \cap N(w_i)$, X is the full set of embeddings, and $X_{\text{projected}} \leftarrow P(X)$.
2. **RLACE** We implement Relaxed Linear Adversarial Concept Erasure (R-LACE; (Ravfogel et al., 2022a)) which defines a linear minimax game to adversarially identify and remove a linear bias subspace. In this approach, $\mathcal{P}_k$ is defined as the set of all $D \times D$ orthogonal projection matrices that neutralize a rank r subspace:
$$P \in \mathcal{P}_k \leftrightarrow P = I_D - W^\top W$$
The minimax equation is then solved to obtain the projection matrix P which is used to calculate the final intervened embedding $X_{\text{projected}}$, similar to INLP
$$\min_{\theta \in \Theta} \max_{P \in \mathcal{P}_k} \sum_{n=1}^{N} \ell \left(y_n, g^{-1} \left(\theta^\top P x_n \right) \right)$$
Hyperparameters for P and θ were a learning rate of 0.005 and weight decay of 1e-5.

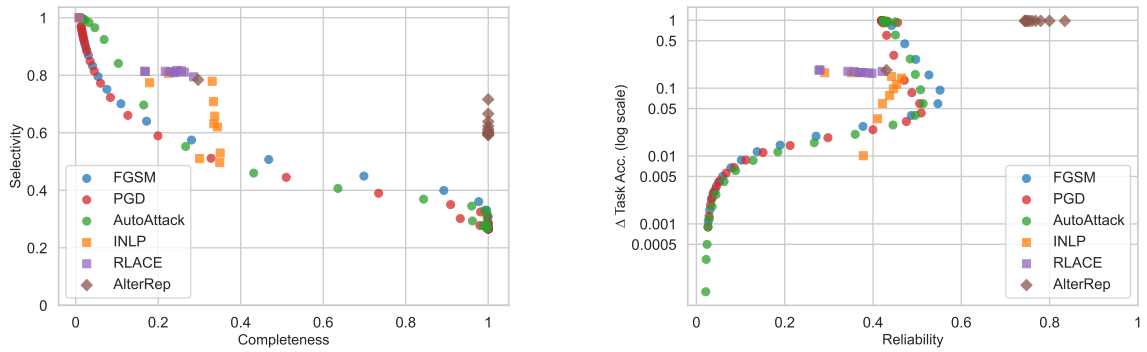
AlterRep We implement AlterRep (Ravfogel et al., 2021) by first running INLP, saving all classifiers, and using these to compute rowspace projections that push all embeddings to the positive $Z = \texttt{Pl}$ or negative $Z = \texttt{Sg}$ side of the separating hyperplane for all classifiers. That is, we compute
$$\hat{\mathbf{h}}^l_{Z=\texttt{Sg}} = P_N(\mathbf{h}) + \alpha \sum_{w \in \mathbf{W}} (-1)^{SIGN(w \cdot \mathbf{h})} (w \cdot \mathbf{h}) \mathbf{h}$$
$$\hat{\mathbf{h}}^l_{Z=\texttt{Pl}} = P_N(\mathbf{h}) + \alpha \sum_{w \in \mathbf{W}} (-1)^{1-SIGN(w \cdot \mathbf{h})} (w \cdot \mathbf{h}) \mathbf{h}$$
where P_N is the nullspace projection from INLP.

C Supplemental Results

C.1 Final-Layer Completeness, Reliability, and Selectivity

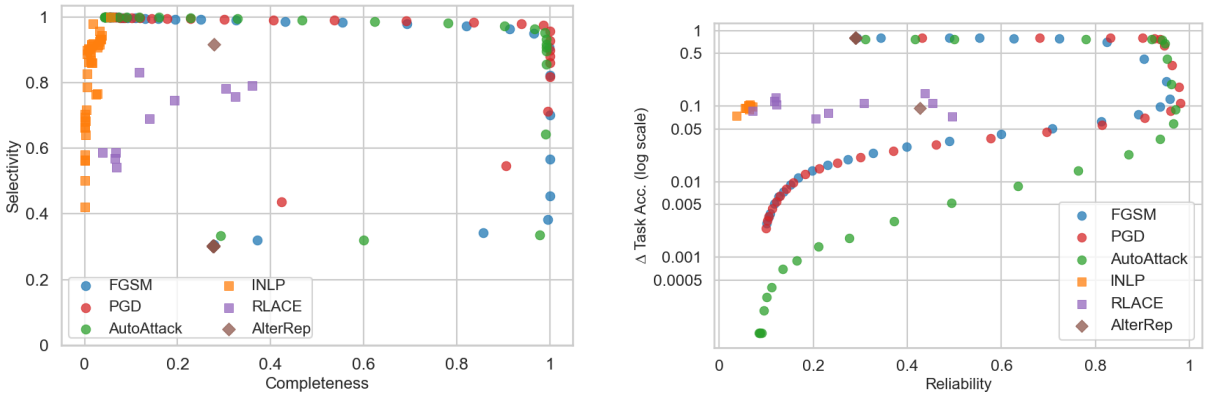
In Figures 4a, 5a and 6a to 6c, we visualize final-layer completeness and selectivity of intervention methods for all models except Pythia-160M, analogously to the Figure 2a results reported in the main paper for Pythia-160M. In Figures 4b and 5b, we show the relationship between ΔTask Acc and reliability for BERT and GPT2 (respectively), analogously to the Figure 2b results reported in the main paper for Pythia-160M.

[8] This is only true for binary properties Z – for variables that can take n values with $n > 2$, the number of dimensions removed is $n \cdot r$.



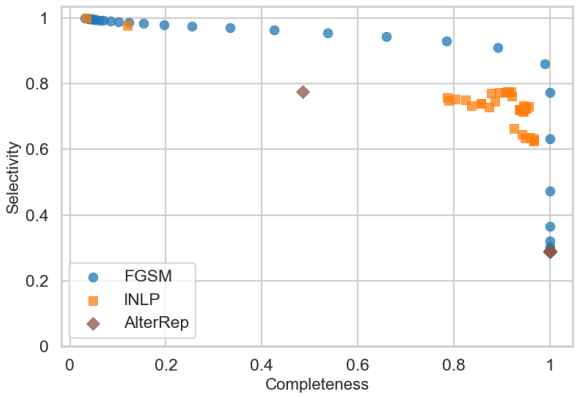
(a) **Selectivity vs. Completeness** (b) **ΔTask Acc vs. Reliability**

Figure 4: **(BERT) Completeness, selectivity, reliability, and ΔTask Acc** for all interventions in BERT's final layer. Each point in both plots corresponds to a different hyperparameter setting.

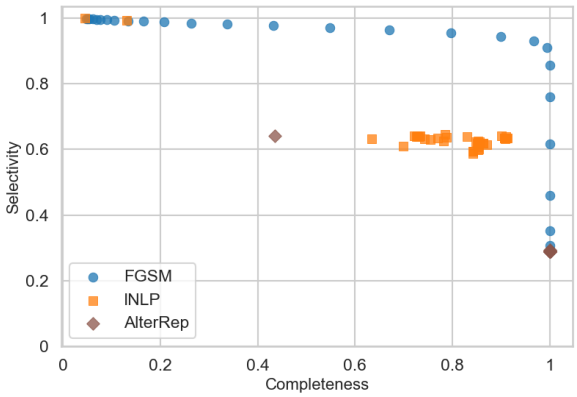


(a) **Selectivity vs. Completeness** (b) **ΔTask Acc vs. Reliability**

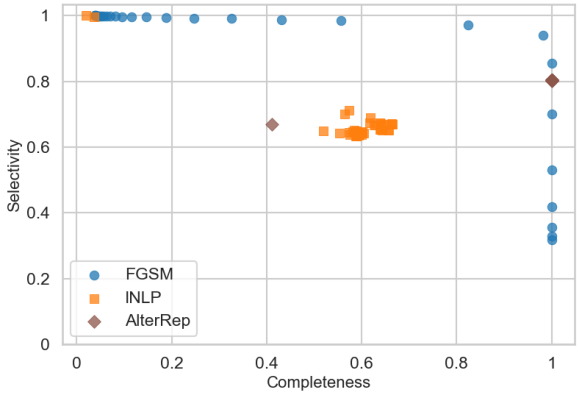
Figure 5: **(GPT2) Completeness, selectivity, reliability, and ΔTask Acc** for all interventions in the final layer of GPT2. Each point in both plots corresponds to a different hyperparameter setting.



(a) **(Pythia-1.4B) Selectivity vs. Completeness**



(b) **(Pythia-6.9B) Selectivity vs. Completeness**



(c) **(Llama-3.2-3B-Instruct) Selectivity vs. Completeness**

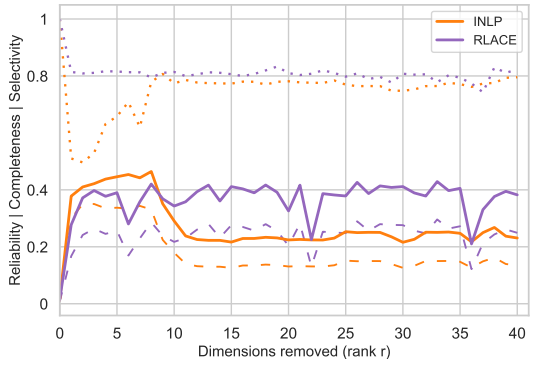
Figure 6: **Completeness and selectivity** for all interventions in the final layer of **Pythia-1.4B, Pythia-6.9B, and Llama-3.2-3B-Instruct**. Each point in both plots corresponds to a different hyperparameter setting.

	$C(\hat{\mathbf{H}}_Z)$	$S(\hat{\mathbf{H}}_Z)$	$R(\hat{\mathbf{H}}_Z)$	x_{opt}
INLP	0.3308 ± 0.0013	0.7792 ± 0.0012	0.4644 ± 0.0013	$r = 8$
RLACE	0.2961 ± 0.0013	0.7782 ± 0.0012	0.4290 ± 0.0014	$r = 33$
AlterRep	1.0000 ± 0.0000	0.7162 ± 0.0017	$\mathbf{0.8346} \pm 0.0012$	$\alpha = 0.1$
FGSM	0.8923 ± 0.0011	0.3994 ± 0.0018	0.5518 ± 0.0017	$\varepsilon = 0.112$
PGD	0.7343 ± 0.0016	0.3897 ± 0.0018	0.5092 ± 0.0016	$\varepsilon = 0.112$
AutoAttack	0.8433 ± 0.0013	0.3692 ± 0.0019	0.5136 ± 0.0018	$\varepsilon = 0.112$

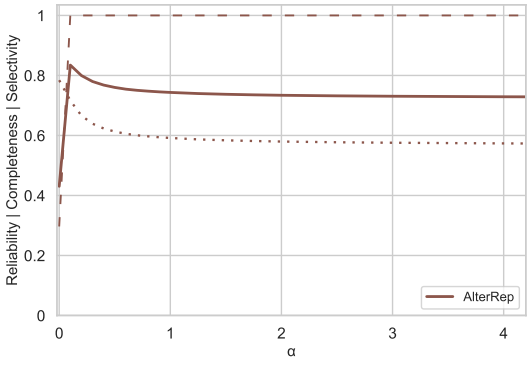
Table 3: **(BERT) Intervention scores for maximum-reliability hyperparameters** in the final layer, **with standard error included**. All scores are reported for the hyperparameter x_{opt} that maximizes the reliability of each respective method. Counterfactual methods are grouped above the double line, with concept removal methods below it.

Additionally, in Table 3, we report the standard error of completeness, selectivity, and reliability for BERT's maximum-reliability final-layer results displayed in Table 1. Note that all scores have standard error < 0.002, and we observe the same pattern for all other models.

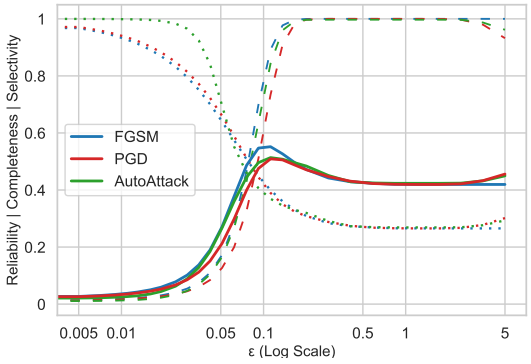
Hyperparameter Variation In Figures 7 and 8, we observe that increasing the degree of control that interventions have over the representation of the target property by increasing the intervention hyperparameter associated with a given intervention type (i.e., ε, α, or rank) generally leads to both improved completeness and decreased selectivity.



(a) **Linear Removal (INLP, RLACE)**

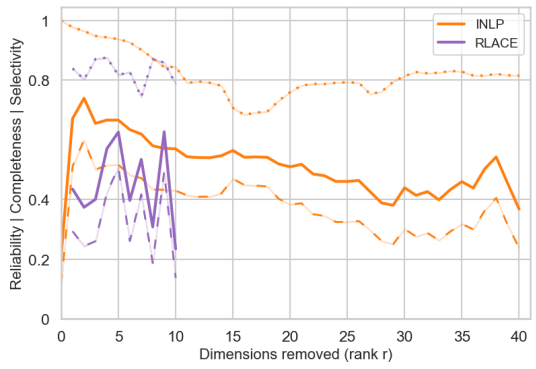


(b) **Linear Counterfactual (AlterRep)**

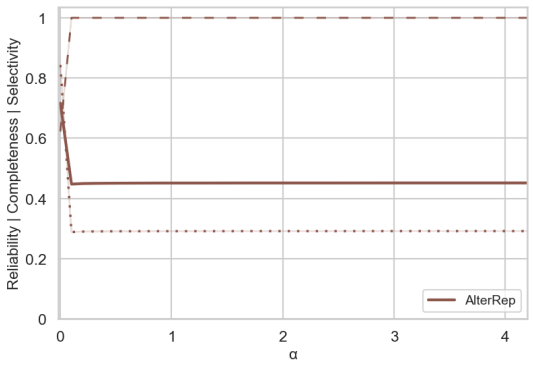


(c) **Nonlinear Counterfactual (GBIs)**

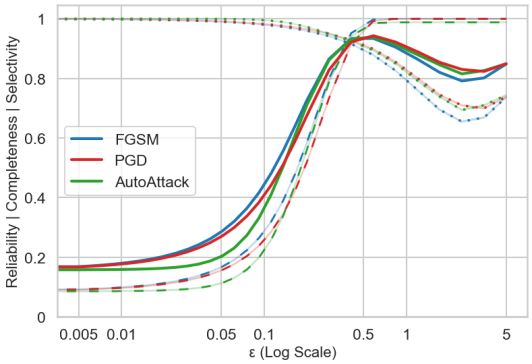
Figure 7: **(BERT) Reliability (solid), completeness (dashed), & selectivity (dotted)** of all methods in BERT's final layer, by hyperparameter value.



(a) **Linear Removal (INLP, RLACE)**



(b) **Linear Counterfactual (AlterRep)**

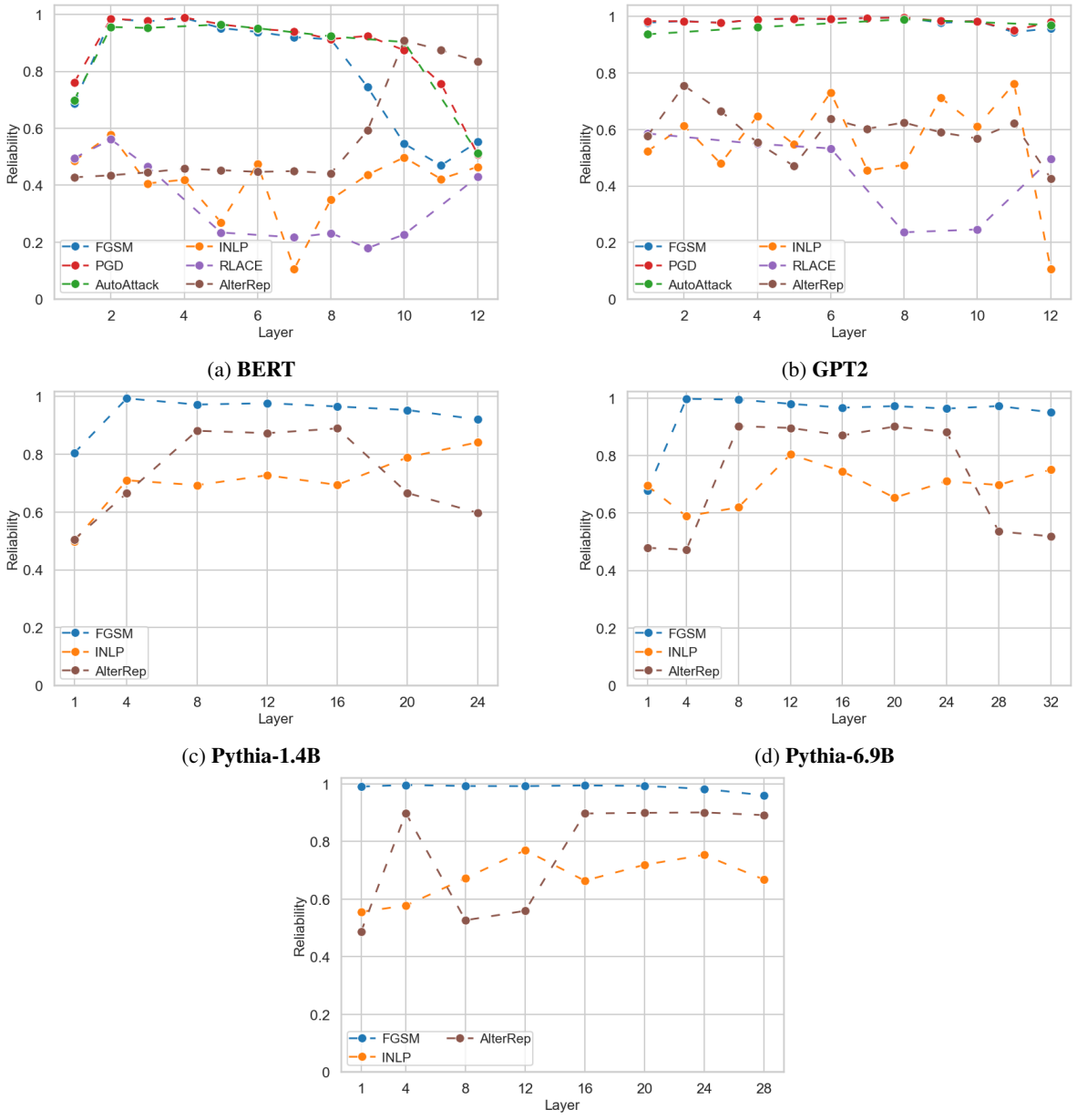


(c) **Nonlinear Counterfactual (GBIs)**

Figure 8: **(Pythia-160M) Reliability (solid), completeness (dashed), & selectivity (dotted)** of all methods in the final layer of Pythia-160M, by hyperparameter value.

C.2 Reliability by Layer

In Figure 9, we visualize maximum reliability of intervention methods across layers for all models (except Pythia-160M, which is reported in the main paper), analogously to the Figure 3 results reported in Section 5.2.



(a) **BERT**

(b) **GPT2**

(c) **Pythia-1.4B**

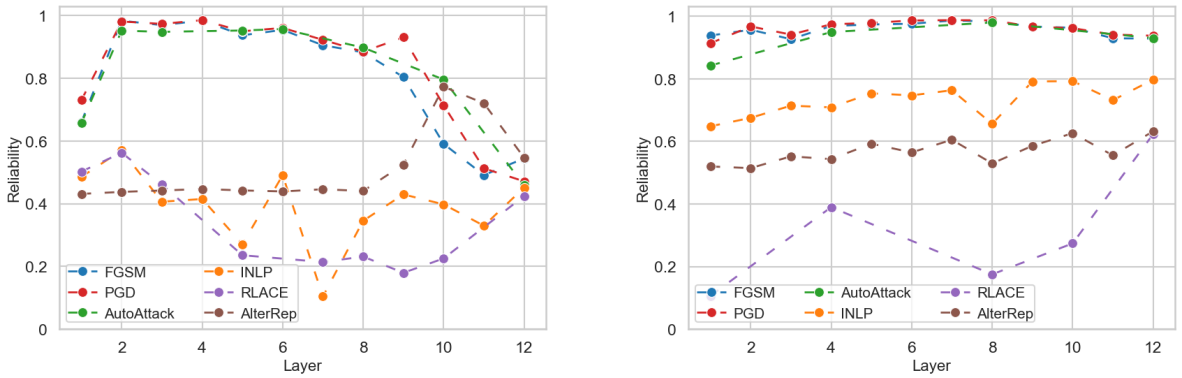
(d) **Pythia-6.9B**

(e) **Llama-3.2-3B-Instruct**

Figure 9: **Maximum reliability by layer** across models for each intervention across all layers.

C.3 Linear Validation Probes

We present the reliability by layer for BERT and Pythia-160M using *linear validation probes* in Figures 10a and 10b (respectively). The main trends (specifically, reliability ordering of methods by layer) shown here are very similar to those using MLP validation probes, as shown in Figures 3 and 9a, with the exception that the linear counterfactual method (AlterRep) does not surpass the reliability of GBIs as strongly in the later layers (for BERT). The BERT result is not especially surprising, as linear validation probes are expected to be less resilient to linear interventions than MLP validation probes (as MLPs can also rely on nonlinearly-encoded information to make predictions) leading to lower selectivity and correspondingly lower reliability using linear interventions with linear validation probes compared to evaluations using nonlinear validation probes. However, it is important to note that the overall ordering of methods, and the specific scores observed, are still remarkably similar between linear vs nonlinear validation probes for both models, indicating that the differences in reliability between linear and nonlinear methods are unlikely to be due to the (non)linearity of validation probes.

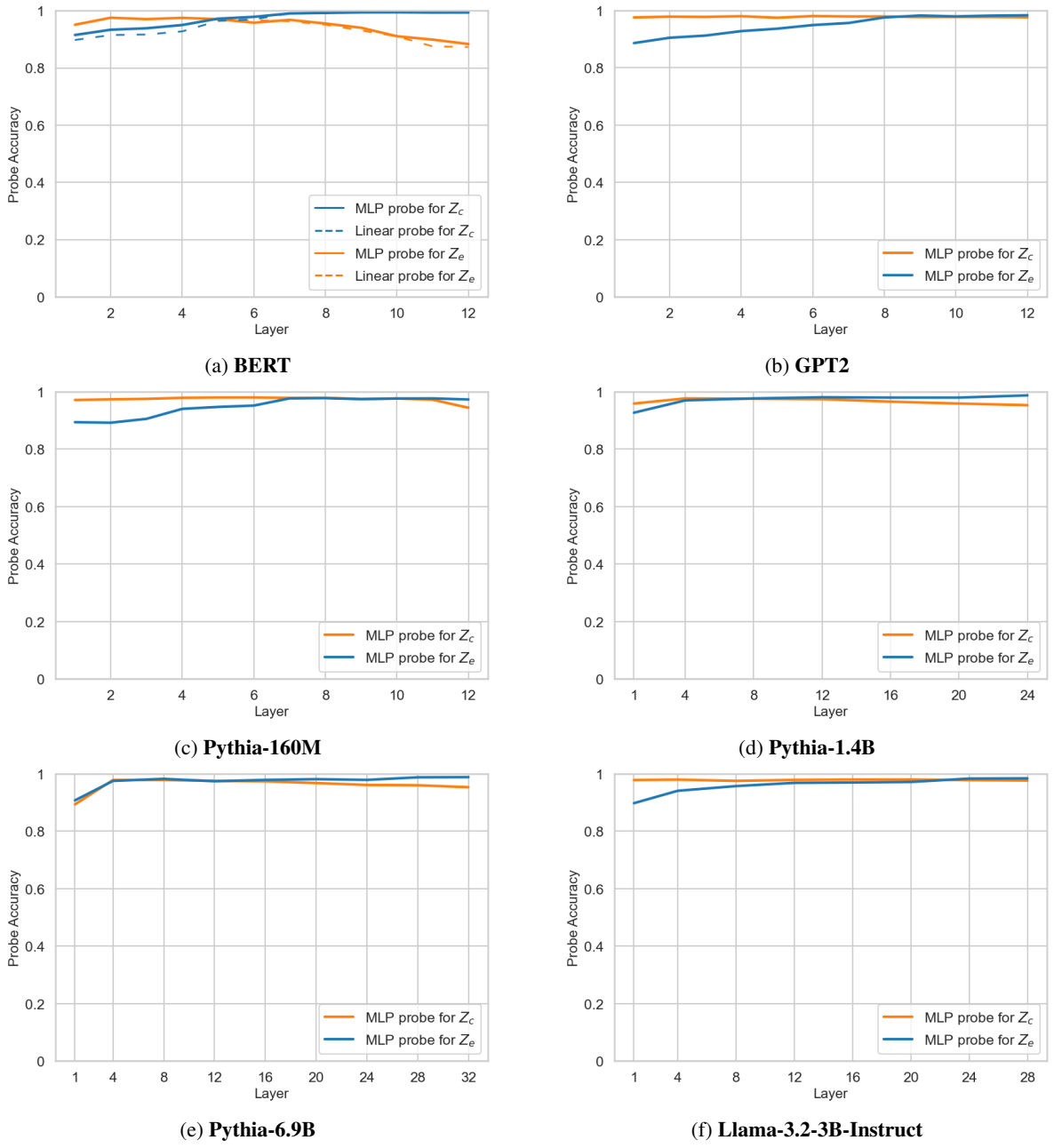


(a) **BERT** (b) **Pythia-160M**

Figure 10: **Maximum reliability by layer** for each intervention across all layers, using *linear validation probes*.

C.4 Validation Probe Accuracy by Layer

In Figure 11, we report the layerwise validation probe accuracy across all models.



(a) **BERT** (b) **GPT2**

(c) **Pythia-160M** (d) **Pythia-1.4B**

(e) **Pythia-6.9B** (f) **Llama-3.2-3B-Instruct**

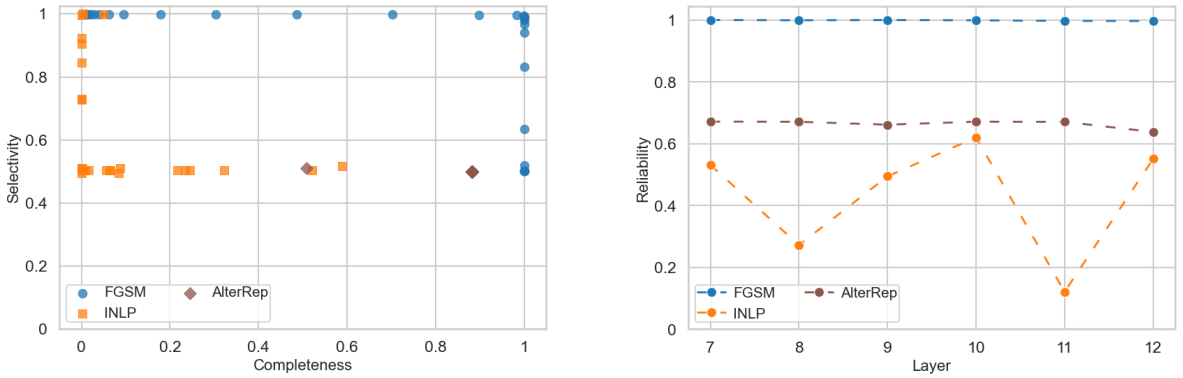
Figure 11: **Validation Probe Accuracy by Layer** across models.

C.5 Results on Indirect Object Identification (IOI) Task

The Indirect Object Identification (IOI) task (Wang et al., 2023) is a well-studied benchmark in mechanistic interpretability research, requiring models to identify the correct referent in sentences with multiple names. Each sentence has an initial dependent clause introducing two names (e.g., "After John and Mary went to the store..."), followed by a main clause where one person performs an action involving the other (e.g., "... John gave a bottle of milk to..."). The model must correctly complete the sentence with "Mary" (i.e., the indirect object) rather than repeating "John".

For our experiments, we define the causal variable Z_c as denoting whether the first or second mentioned person in the sequence is the correct indirect object (each label has probability 0.5 in this dataset), and the environmental variable Z_e as the tense of the root verb (e.g., "gives" vs "gave"), which is clearly irrelevant to solving the task.

Following Zhang and Nanda (2024), we study IOI in the context of GPT2. Figure 12 shows the relation between selectivity and completeness as well as the reliability of the various interventions across layers. (Note that we only display results on layers 7–12 because the earlier layers do not encode necessary information to predict the variables of interest – i.e., in these layers, we cannot train a probe to predict the target features at sufficiently high accuracy, as also observed by Zhang and Nanda 2024.) The results are similar with the subject-verb agreement task on GPT2 in that FGSM (nonlinear) is more reliable than INLP and AlterRep (linear) at all layers; but on the IOI task, we find that AlterRep is more reliable than INLP at all layers (for subject-verb agreement, these methods' relative performances varied on GPT2).



(a) **Selectivity vs. Completeness** (b) **Reliability by Layer**

Figure 12: **Results on Indirect Object Identification (IOI) task.** Completeness, selectivity, and reliability for interventions using the IOI task and the GPT2 model.