

When LRP Diverges from Leave-One-Out in Transformers

Weiqiu You[†] Siqi Zeng[‡] Yao-Hung Hubert Tsai[§] Makoto Yamada[§] Han Zhao[‡]

[†]University of Pennsylvania, Philadelphia, PA, USA

[‡]University of Illinois at Urbana-Champaign, Urbana, IL, USA

[§]Okinawa Institute of Science and Technology, Okinawa, Japan

Abstract

Leave-One-Out (LOO) provides an intuitive measure of feature importance but is computationally prohibitive. While Layer-Wise Relevance Propagation (LRP) offers a potentially efficient alternative, its axiomatic soundness in modern Transformers remains largely under-examined. In this work, we first show that the bilinear propagation rules used in recent advances of AttnLRP violate the implementation invariance axiom. We prove this analytically and confirm it empirically in linear attention layers. Second, we also revisit CP-LRP as a diagnostic baseline and find that bypassing relevance propagation through the softmax layer—backpropagating relevance only through the value matrices—significantly improves alignment with LOO, particularly in middle-to-late Transformer layers. Overall, our results suggest that (i) bilinear factorization sensitivity and (ii) softmax propagation error potentially jointly undermine LRP’s ability to approximate LOO in Transformers.¹

1 Introduction

As Transformer-based machine learning models become central to high-stakes domains like healthcare (Tjoa and Guan, 2019; Hameed et al., 2023) and legal systems (Zeng et al., 2016; Wexler, 2017), the need for faithful explanations is critical. In particular, unfaithful explanations risk misleading domain experts, thereby undermining their ability to make informed decisions.

An intuitive but computationally prohibitive explanation method is Leave-One-Out (LOO), which measures the impact of removing each feature on a model’s prediction. To approximate LOO efficiently, AttentionRollout (Abnar and Zuidema, 2020) was introduced to move beyond simplistic single-layer attention scores by composing them

¹Correspondence to weiqiuy@seas.upenn.edu. Code is available at https://github.com/fallcat/attn_loo.

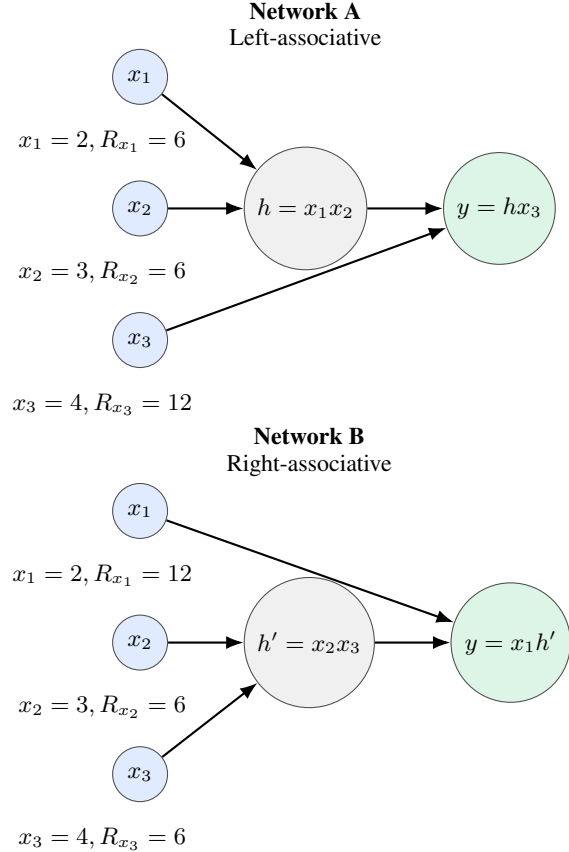


Figure 1: Two functionally equivalent network factorizations of the same function $y = x_1 x_2 x_3$. Despite identical outputs, LRP’s ε -rule assigns different relevance to inputs depending on factorization.

across layers, offering a more intuitive way to track information flow. However, these methods are incomplete as they propagate only attention scores while ignoring other key components like values and hidden states.

Layer-Wise Relevance Propagation (LRP) (Bach et al., 2015) redistributes a model’s output score backward through the network while conserving relevance across layers, offering a more principled framework for approximating LOO. Recent advances in LRP for attention rules in Transform-

ers (Vaswani et al., 2017) include CP-LRP (Ali et al., 2022) and AttnLRP (Achtibat et al., 2024). CP-LRP regards the matrix multiplication between attention weights and value as a linear layer, propagating relevance scores only through the value vectors (Ali et al., 2022). AttnLRP critiques CP-LRP for being unfaithful by ignoring the propagation through the attention weights, and proposes a new rule for bilinear layer using deep Taylor decomposition (DTD), which improves upon insertion and deletion related metrics (Achtibat et al., 2024).

However, previous works have shown that LRP in general violates *implementation invariance* axiom—the principle that functionally identical networks should yield identical explanations (Sundararajan et al., 2017). This flaw was first shown in a counterexample where two networks, differing only in their internal arrangement of ReLU operations, produced different LRP attributions despite being functionally equivalent (Sundararajan et al., 2017). With the recent CP-LRP and AttnLRP, it has not yet been shown whether this flaw still exists or whether these LRP variants can approximate LOO in Transformers well despite these limitations.

In this work, we provide the first formal proof that LRP’s axiomatic failure is not a rare corner case but still exists in AttnLRP’s new rule for bilinear layers. We establish this through both theory and experiments: first, by proving the violation with a simple analytical example, and then by empirically demonstrating it in a one-layer linear attention model. Moreover, we empirically find that, although CP-LRP lags behind AttnLRP in insertion and deletion based metrics, CP-LRP is actually a better approximation for LOO. Ablating their application layer by layer, we find that treating attention weights as constants and bypassing softmax-layer backpropagation (as in CP-LRP) improves LOO alignment in middle-to-late layers, whereas AttnLRP’s bilinear propagation can be more beneficial in earlier layers. This is observed in our layerwise analysis (Section 4), where several middle-to-late layers exhibit flatter attention distributions.

Our contributions are twofold and relate to different parts of attention mechanisms:

Error in Bilinear Layers. To the best of our knowledge, we are the first to formally prove that AttnLRP’s propagation rule for bilinear layers violates the implementation invariance axiom. We identify a fundamental source of LOO estimation

error that extends beyond previously studied nonlinearities to the core bilinear operations underlying modern attention mechanisms. Using a small linear attention model trained on MNIST (LeCun, 1998), we empirically confirm that LRP attributions differ between left- and right-associative bilinear factorizations: their scores are not fully correlated and exhibit inconsistent alignment with LOO.

Error in Softmax Layers. We compare the two Transformer-based LRP variants, CP-LRP (LRP for Transformers without softmax propagation) and AttnLRP (LRP for Transformers with bilinear and softmax propagation), and show that CP-LRP correlates with LOO better in BERT (Devlin et al., 2019) on SST (Socher et al., 2013) and IMDB (Maas et al., 2011). We conduct a layer-wise ablation and show that regarding the attention weights as constant like CP-LRP in the middle-to-late layers in Transformers is the most helpful for approximating LOO.

2 Background

In this section, we build the foundation for our analysis by first establishing the desired properties of an attribution method and then detailing the mechanics of LRP as a popular approximation. We begin by formally defining LOO attribution, establishing it as a conceptual benchmark due to its faithfulness and implementation invariance. As LOO has a prohibitive computational cost, there is a pressing need for efficient alternatives. We then introduce LRP, a principled, single-pass method designed to overcome the limitations of both LOO and early heuristics. We will describe its core principles of relevance conservation and layer-wise decomposition before examining its specific adaptations for the Transformer architecture, setting the stage for our theoretical critique.

2.1 Leave-One-Out (LOO) Attribution

A central challenge in attribution evaluation is identifying a reliable reference metric that reflects the true contribution of each input feature. The LOO score provides such a reference by measuring the change in the model’s output when a single feature is removed. Formally, for an input x and feature i , the LOO score is calculated as:

$$\text{LOO}_i = f(x) - f(x_{\setminus i}) \quad (\text{LOO})$$

where $x_{\setminus i}$ denotes the input with feature i removed. Because this definition depends only on the model’s

functional behavior, the abstract rule that defines an operation as a mapping from inputs to outputs—rather than its specific implementation—the particular procedure to realize that mapping, LOO, is invariant to reparameterizations and satisfies the **implementation invariance** axiom. LOO is widely regarded as a common baseline for feature importance (Ancona et al., 2018; Covert et al., 2021). However, computing LOO scores exactly requires one forward pass per feature, which is computationally prohibitive for modern deep models. This motivates the development of scalable attribution methods that aim to approximate LOO scores efficiently while preserving their desirable properties.

2.2 Heuristic LOO approximations

Early attempts to approximate LOO in Transformers focused on the attention mechanism itself, as attention weights provide an intuitive story for how information is combined across the input. Methods like Attention Rollout (Abnar and Zuidema, 2020) sought to address the limitation of using raw, single-layer attention weights by composing them across layers. The intuition is to recursively propagate attention scores from the final layer to the input. The effective attention from layer l down to the input is computed by recursively multiplying attention matrices:

$$\tilde{A}^{(l)} = A^{(l)} \cdot \tilde{A}^{(l-1)} \quad (\text{Rollout})$$

where $A^{(l)}$ is the attention matrix at layer l and the rollout begins with an identity matrix.

However, such heuristic methods are incomplete approximations of LOO. By focusing exclusively on attention matrices (A), they ignore the contributions of other critical components, such as the value projections (V), feed-forward networks, and residual connections. The true final prediction is a function of the entire computation path, not just the attention patterns. The significance of this limitation becomes clear when considering findings that Transformer performance can be surprisingly robust even when learned attention is replaced with hard-coded, non-data-dependent patterns (You et al., 2020). This suggests that much of the model’s predictive power is encoded in the value transformations and subsequent layers—a significant part of the model that attention-only methods completely disregard.

2.3 A Principled LOO Alternative: Layer-Wise Relevance Propagation (LRP)

To overcome the shortcomings of heuristic methods, more principled approaches like LRP have been adapted for Transformers. LRP offers a complete decomposition of the model’s prediction by propagating relevance backward from the output to the input², accounting for *every component* in the computational graph. It operates on a conservation principle, where the model’s output score $f(x)$ is decomposed into a sum of relevance scores R_i for each input feature, such that $f(x) = \sum_i R_i$. This is achieved by propagating the total relevance backward through the network, conserving it at each layer.

Epsilon Rule for Linear Layers. For standard linear layers, a common propagation choice is the ε -rule. This rule distributes relevance from a neuron to its inputs in proportion to their contribution to its activation, a principle rooted in the Deep Taylor Decomposition (DTD) framework:

$$R_i^{(l-1)} = \sum_j \frac{z_{ij}^{\text{lin}}}{\sum_k z_{kj}^{\text{lin}} + \varepsilon \cdot \text{sign}(\sum_k z_{kj}^{\text{lin}})} R_j^{(l)}, \quad (1)$$

where $z_{ij}^{\text{lin}} = x_i^{(l-1)} W_{ij}$ represents the contribution of input neuron i to neuron j . In the DTD framework, this corresponds to the input multiplied by the partial derivative (the weight W_{ij}), forming the basis for distributing relevance. ε is a small positive constant added for numerical stability.

Rules for Bilinear Layers. We make the attention forward pass explicit:

$$\begin{aligned} Z &= \frac{1}{\sqrt{d_k}} QK^\top, \\ A &= \text{softmax}(Z) \text{ (row-wise)}, \\ O &= AV. \end{aligned} \quad (2)$$

Bilinearity exists in two places: $O = AV$ and $Z = QK^\top / \sqrt{d_k}$.

CP-LRP (Ali et al., 2022) treats $O = AV$ as linear in V (holding A fixed), backpropagating relevance only through V while A (and Q , K as well) gets zero relevance scores. **AttnLRP** (Achibat et al., 2024) instead uses DTD (Montavon et al., 2017) to split relevance between both factors. With a small stabilizer $\varepsilon > 0$:

²LRP methods only involve backpropagation, while the forward pass is just one standard forward pass from the input to the final output of the Transformer.

Incoming relevance at O_{jp} (denoted $R_{jp}^{(l)}$) is split to A and V as

$$R_{A_{ji}}^{(l-1)} = \sum_p \frac{A_{ji} V_{ip}}{2 O_{jp} + \varepsilon \text{sign}(O_{jp})} R_{jp}^{(l)}, \quad (3a)$$

$$R_{V_{ip}}^{(l-1)} = \sum_j \frac{A_{ji} V_{ip}}{2 O_{jp} + \varepsilon \text{sign}(O_{jp})} R_{jp}^{(l)}. \quad (3b)$$

Relevance is split in half assigning the same value to two values that multiply together inside matrices in bilinear operations. The sum of relevance in attention A is the same as the sum in value V : $R_A^{(l-1)} = R_V^{(l-1)} = \frac{1}{2} R_O^{(l)}$, in the limit $\varepsilon \rightarrow 0$.

Similarly, the relevance score from the pre-softmax attention logits $Z = QK^\top / \sqrt{d_k}$, where we define $c_{jir} \triangleq \frac{Q_{jr} K_{ir}}{\sqrt{d_k}}$ so that $Z_{ji} = \sum_r c_{jir}$, can be split to Q and K as

$$R_{Q_{jr}}^{(l-1)} = \sum_i \frac{c_{jir}}{2 Z_{ji} + \varepsilon \text{sign}(Z_{ji})} R_{Z_{ji}}^{(l)}, \quad (4a)$$

$$R_{K_{ir}}^{(l-1)} = \sum_j \frac{c_{jir}}{2 Z_{ji} + \varepsilon \text{sign}(Z_{ji})} R_{Z_{ji}}^{(l)}. \quad (4b)$$

The sum of relevance in query Q is the same as the sum in key K : $R_Q^{(l-1)} = R_K^{(l-1)} = \frac{1}{2} R_Z^{(l)}$, in the limit $\varepsilon \rightarrow 0$.

These AttnLRP bilinear rules are used with the softmax rule in the next subsection. For the whole backward propagation process, relevance is first split at $O \rightarrow (A, V)$ via Equation (3), then passed through $A = \text{softmax}(Z)$ (Equation (5)), and finally split at $Z \rightarrow (Q, K)$ via Equation (4).

Rule for the Softmax Layer. AttnLRP also derives rules for softmax layers using DTD. Let Z be the logits and $A = \text{softmax}(Z)$ the attention weights. We write superscripts $(l), (l-1)$ for the layers indicating propagation direction and use subscripts to denote tensors (Z, A, V) . Following Achibat et al. (2024), a first-order Taylor expansion leads to the element-wise rule

$$R_{Z_{ji}}^{(l-1)} = Z_{ji} \left(R_{A_{ji}}^{(l)} - A_{ji} \sum_{i'} R_{A_{ji'}}^{(l)} \right), \quad (5)$$

$$A_{ji} = \text{softmax}(Z_{\cdot j})_i,$$

which is applied entry-wise over (j, i) to Z and A . Rule Equation (5) moves relevance from A back to the logits Z ; from there, the bilinear rule for Z distributes relevance to Q and K (see Equation (4)). Together with Equation (3), these rules conserve relevance locally while making explicit which components (attention weights vs. values vs. logits) carry the propagated mass.

Since CP-LRP treats bilinear layers as linear, relevance is not propagated through A , and consequently, the softmax propagation rule—which depends on relevance scores from the attention—does not apply.

While these rules are derived from a principled framework and satisfy local relevance conservation, they rely on intermediate activations from the forward pass (e.g. z_{ij} in the ε -rule, which is computed using the input activation from the previous layer, and O_{jp} in the bilinear rule), making them *sensitive to the precise order of computations*. This stands in contrast to the associative property found in standard arithmetic, which will cause failure for approximating LOO. This sensitivity to computation order hints at a deeper limitation of LRP: even in simple multiplicative settings, its attributions can depend on implementation details rather than functional behavior, which we formalize next.

3 Why does LRP still approximate LOO poorly?

While LRP’s layer-local propagation rules guarantee relevance conservation, they do not guarantee consistent explanations for functionally equivalent networks. We first analyze how this axiomatic failure arises from the bilinear operations at the core of Transformer attention, formally proving that AttnLRP’s propagation rule is sensitive to the factorization of these operations. To probe the second potential source of error, we compare AttnLRP to a CP-LRP which bypasses the softmax step, allowing us to isolate softmax layer’s impact from bilinear operations on LOO correlation. Taken together, these two perspectives—on bilinear factorization and softmax propagation—reveal fundamental weaknesses in current LRP variants as approximations to LOO.

3.1 Part 1: Bilinear Propagation in LRP Violates Implementation Invariance

Implementation invariance requires that two networks computing the same function produce identical explanations (Sundararajan et al., 2017). While methods like Integrated Gradients satisfy this axiom by design, propagation-based methods such as LRP can violate it. Earlier works demonstrate that LRP is not implementation invariant when applying its propagation rules to ReLU and BatchNorm layers (Sundararajan et al., 2017; Guillemot et al., 2020; Yeom et al., 2021). We extend this analysis

to **bilinear operations**, a core component of Transformer attention, and show that the ε -rule for bilinear layers introduced in AttnLRP (Achtibat et al., 2024) are sensitive to the computational *factorization* of these operations, even when the underlying function remains identical.

We demonstrate this flaw with a simple scalar example. Consider the function $f(x_1, x_2, x_3) = x_1 x_2 x_3$, which can be implemented in two computationally equivalent ways:

- **Model A (Left-associative):** $y = (x_1 x_2) x_3$
- **Model B (Right-associative):** $y = x_1 (x_2 x_3)$

As illustrated in Figure 1, let $x_1 = 2, x_2 = 3, x_3 = 4$, yielding an output $y = 24$. If we assign this output as the total relevance, $R_y = 24$, the standard LRP rule for multiplication (equal splitting) distributes relevance at each step.

Model A Derivation The relevance is propagated backward as follows:

- First, the relevance $R_y = 24$ is split between $(x_1 x_2)$ and x_3 , so each receives 12.
- Then, the 12 assigned to $(x_1 x_2)$ is split between x_1 and x_2 , so each receives 6.

This yields final relevance scores of:

$$R_{x_1} = 6, \quad R_{x_2} = 6, \quad R_{x_3} = 12$$

Model B Derivation In contrast, the right-associative model’s propagation is:

- First, the relevance $R_y = 24$ is split between x_1 and $(x_2 x_3)$, so each receives 12.
- Then, the 12 assigned to $(x_2 x_3)$ is split between x_2 and x_3 , so each receives 6.

This results in different final relevance scores:

$$R_{x_1} = 12, \quad R_{x_2} = 6, \quad R_{x_3} = 6$$

The relevance scores for x_1 and x_3 are swapped based on the grouping of operations, even though the function, output, and gradients are identical. In sharp contrast, the LOO scores for both implementations are identical. Setting any single input to zero makes the final output zero, causing a change of 24 from the original output ($LOO_1 = 24, LOO_2 = 24, LOO_3 = 24$). This is because LOO depends only on functional behavior, not on the specific parameterization.

This discrepancy not only exists in this specific example. More generally, any operation that does not satisfy the associative property can lead to implementation-dependent differences in LRP attributions. For instance, in linear attention variants that omit the softmax, the term $QK^\top V$ becomes associative and can be computed as either $(QK^\top)V$ or $Q(K^\top V)$ (Katharopoulos et al., 2020). Although the softmax activation in standard attention obscures the underlying associativity, our analysis of two bilinear factorizations suggests that standard attention may also violate implementation invariance for certain mathematical operations.

3.2 Part 2: Softmax Propagation as a Second Source of Error

In addition to bilinear operations, as another key component of the attention mechanism, we further hypothesize that the propagation rules in softmax layers are also problematic. There are at least two types of error introduced by softmax propagation.

(1) Structural bias: when the logits are uniform, the softmax layer outputs a nonzero, uniform attention distribution that reflects a *default behavior* rather than input-dependent evidence. LRP then spreads relevance evenly across inputs, whereas ground-truth LOO would assign near-zero relevance to each feature if all features are actually close to 0.

(2) Linearization error: the softmax propagation rule relies on a first-order Taylor expansion (Equation (5)) around the observed logits. When the logits are large but similar, the attention distribution appears flat even though individual inputs can strongly influence the output. In this regime, the local Jacobian provides little discriminative signal, and the linearization fails to capture the large non-local effect of removing individual inputs—leading to systematic misallocation of relevance.

To isolate the effect of softmax from bilinear operations in LRP, we revisit **CP-LRP** as a diagnostic baseline against AttnLRP. CP-LRP treats the value readout $O = AV$ as linear in V while holding A fixed (Section 2), thereby bypassing propagation through both the attention weights and, particularly, the softmax layer $A = \text{softmax}(Z)$. This makes CP-LRP a natural baseline for isolating the impact of softmax propagation: any gap between CP-LRP and AttnLRP on LOO reflects the added effect of the softmax rule and the bilinear split into A (Equations (3) and (5)). If CP-LRP shows higher LOO agreement, this supports the claim that softmax

propagation can be a major source of attribution error.

4 Experiments

We empirically evaluate these two error sources through three research questions:

- **RQ1:** Does LRP’s ε -rule for bilinear layers produce different attributions for functionally equivalent factorizations (i.e., show implementation variance)?
- **RQ2:** If we handle the attention step like CP-LRP (send all relevance at O to V and skip softmax), do the attributions agree better with LOO than AttnLRP?
- **RQ3:** Which layers were affected the most from bypassing softmax propagation in Transformer attention as in CP-LRP?

4.1 Experimental Setup

Tasks and Datasets. We evaluate attribution methods across three complementary settings: (1) a synthetic bilinear setting using the MNIST dataset (LeCun, 1998), (2) two standard text classification benchmarks, SST (Socher et al., 2013) and IMDB (Maas et al., 2011), and (3) BERT-base (Devlin et al., 2019) for realistic Transformer-based evaluation. For MNIST, we resize the images to 14×14 and use them to probe LRP’s behavior in controlled bilinear networks while retaining non-trivial real inputs. For SST and IMDB, we follow standard text preprocessing and fine-tune BERT-base for classification. Implementation details are provided in Appendix B.

Models. For the MNIST experiments, we design two synthetic QKV networks with linear attention without softmax layers, which differ only in how their bilinear computations are factorized (left- vs. right-associative; see Section 3.1). These models are lightweight but allow us to cleanly test implementation invariance on real data. Note that real Transformers insert a softmax layer between bilinear computations, preventing a direct associativity test. For SST and IMDB, we use a standard 12-layer BERT-base encoder.

Evaluation Metrics. Our primary metric is the Pearson correlation (r) between attribution scores and ground-truth feature importance derived from **Leave-One-Out (LOO)**. As a secondary metric, we use the Area Over the Perturbation Curve

(AOPC) for **Insertion and Deletion** (Petsiuk et al., 2018; Samek et al., 2017), which measures how the model’s prediction changes when features are removed in order of importance.

Additionally, we compute two standard perturbation curves: **MoRF** (Most Relevant First) and **LeRF** (Least Relevant First) (Samek et al., 2017; Petsiuk et al., 2018). For MoRF, features are progressively removed in decreasing order of relevance, and the model’s output is recorded after each step; faithful explanations should cause rapid prediction degradation. LeRF performs the same procedure but removes features from least to most relevant, where higher LeRF indicates more faithful explanation. We summarize each curve using the *Area Over the Perturbation Curve* (AOPC) and report LeRF, MoRF, and their difference $\Delta = \text{LeRF} - \text{MoRF}$, which reflects how well the attribution separates relevant from irrelevant features (higher is better).

Details for Pearson Correlation Computation and LOO.

For each example, we compute the Pearson correlation between token- or pixel-level attribution scores and their corresponding LOO scores, then report the mean correlation across the dataset. All attribution maps are normalized to sum to one per example. To compute LOO scores, we remove each feature individually and measure the change in the model’s predicted logit. For text, this is done by masking out each token using the attention mask. For images, we zero out each pixel one at a time.

4.2 Baselines

We evaluate **Integrated Gradients (IG)** (Sundararajan et al., 2017), **Attention Rollout** (Abnar and Zuidema, 2020), **AttnLRP** (Achtibat et al., 2024), and **CP-LRP** (Ali et al., 2022).

4.3 Results and Analysis

(RQ1) The Bilinear Rule in LRP is Not Implementation Invariant. To test whether LRP’s ε -rule for bilinear layers exhibits implementation variance, we construct two *functionally equivalent* QKV linear attention networks that have two bilinear layers and differ only in the order of matrix multiplication. Each consists of three learned projections (W_Q, W_K, W_V), followed by two matrix multiplications and a linear output layer. Neither includes softmax, so the bilinear operation is associative. The only difference is whether the product

is evaluated right-associatively $(QK^\top)V$ or left-associatively $Q(K^\top V)$ as we proposed at the end of Section 3.1. Both networks are trained on 14×14 MNIST with around 85% accuracies, with shared initialization and weights to ensure they compute the same function but have different computational graphs.

We compute attributions using LOO, IG, and AttnLRP, and measure Pearson correlations between left and right models and with LOO (Table 1). LOO is perfectly invariant ($r = 1.0$), and IG also yields perfect left–right agreement, consistent with its axiomatic invariance. AttnLRP, however, is implementation dependent ($r = 0.79$). Compared to LOO, IG shows a negative but nonzero correlation ($r = -0.37$), whereas AttnLRP exhibits near-zero correlation ($r \approx 0.07$ left; $r \approx -0.07$ right), indicating both implementation dependence and poor alignment. These results show that implementation invariance is necessary but not sufficient: IG passes the invariance test but does not reliably align with LOO—sometimes exhibiting negative or later shown near-zero correlations—while AttnLRP fails on both. Additional analysis on formulation of IG is in Appendix A.

(RQ2) Bypassing Softmax Improves Alignment with LOO. To isolate the effect of softmax propagation in AttnLRP, we compare its performance with CP-LRP. As shown in Table 2, CP-LRP substantially improves correlation with LOO on SST over AttnLRP (e.g., $r = 0.52$ on SST versus $r = 0.22$ for AttnLRP). Improvements on IMDB are smaller, potentially because its longer sentences lead to more correlated features. These findings support the claim that the softmax propagation rule is a major source of attribution error.

(RQ3) Identifying the most impacted layers. We localize where softmax propagation contributes most by applying the CP-LRP in three ways: (1) a single layer at a time, (2) cumulatively from the first k layers (front-to-back), and (3) cumulatively from the last k layers (back-to-front). Figure 2 shows that correlation with LOO improves most when modifying **middle and later layers**. Note that applying the identity at *all* layers coincides with CP-LRP’s behavior at the attention step.

On SST, for example, the “Single layer” ablation peaks around layers 6 and 10, while the “Front-to-Back” curve shows a steep rise starting from layer 6. These results indicate that the softmax rule introduces the most attribution error in the model’s

Table 1: **LRP fails implementation invariance and alignment with LOO in bilinear layers.** We compare feature attributions between left- and right-associative QKV bilinear networks that compute the same function. LOO and IG exhibit perfect implementation invariance (L vs $R = 1$), whereas AttnLRP does not. Correlations with LOO are computed separately for the left and right models (identical for LOO by definition), showing that IG has negative but nontrivial correlation whereas AttnLRP is near zero, indicating that invariance alone is not sufficient for faithfulness.

Explainer	L vs R	L vs LOO	R vs LOO
LOO	1.0000	1.0000	1.0000
IG	1.0000	−0.3707	−0.3707
AttnLRP	<u>0.7865</u>	<u>0.0730</u>	<u>−0.0698</u>

Table 2: Baselines on attribution metrics (SST top, IMDB bottom). CP-LRP shows substantially higher agreement with LOO than AttnLRP, highlighting the softmax propagation rule as a source of error. Metrics: LOO r , LeRF, MoRF, and $\Delta = \text{LeRF} - \text{MoRF}$. First place is **bolded** and second place is *italicized* (LOO’s correlation with itself excluded).

Method	LOO $r \uparrow$	LeRF $\uparrow$	MoRF $\downarrow$	$\Delta \uparrow$
SST				
LOO	1.00	1.11	0.33	0.78
IG	−0.05	0.48	0.36	0.12
AttentionRollout	0.08	0.72	0.38	0.35
LRP VARIANTS				
AttnLRP	<i>0.22</i>	0.88	0.05	0.83
CP-LRP	0.52	<i>1.00</i>	<i>0.22</i>	<i>0.79</i>
IMDB				
LOO	1.00	2.20	−0.94	3.14
IG	0.00	1.98	2.70	−0.72
AttentionRollout	0.04	2.85	1.38	1.47
LRP VARIANTS				
AttnLRP	<i>0.25</i>	3.72	<i>−2.64</i>	<i>6.36</i>
CP-LRP	0.26	3.73	−3.02	6.75

middle-to-late layers, where feature interactions are more complex.

Summary of Findings. Our experiments yield three takeaways. (1) In a controlled bilinear setting without softmax, AttnLRP’s ε -rule for bilinear layers violates implementation invariance, assigning different attributions to functionally identical factorizations. (2) The softmax step is a major source of attribution error in Transformers: CP-LRP achieves higher agreement with LOO where on SST, r rises from 0.22 (AttnLRP) to 0.52 (CP-LRP). (3) Layer-wise ablations show that gains concentrate in middle-to-late layers, indicating where softmax propagation harms faithfulness most.

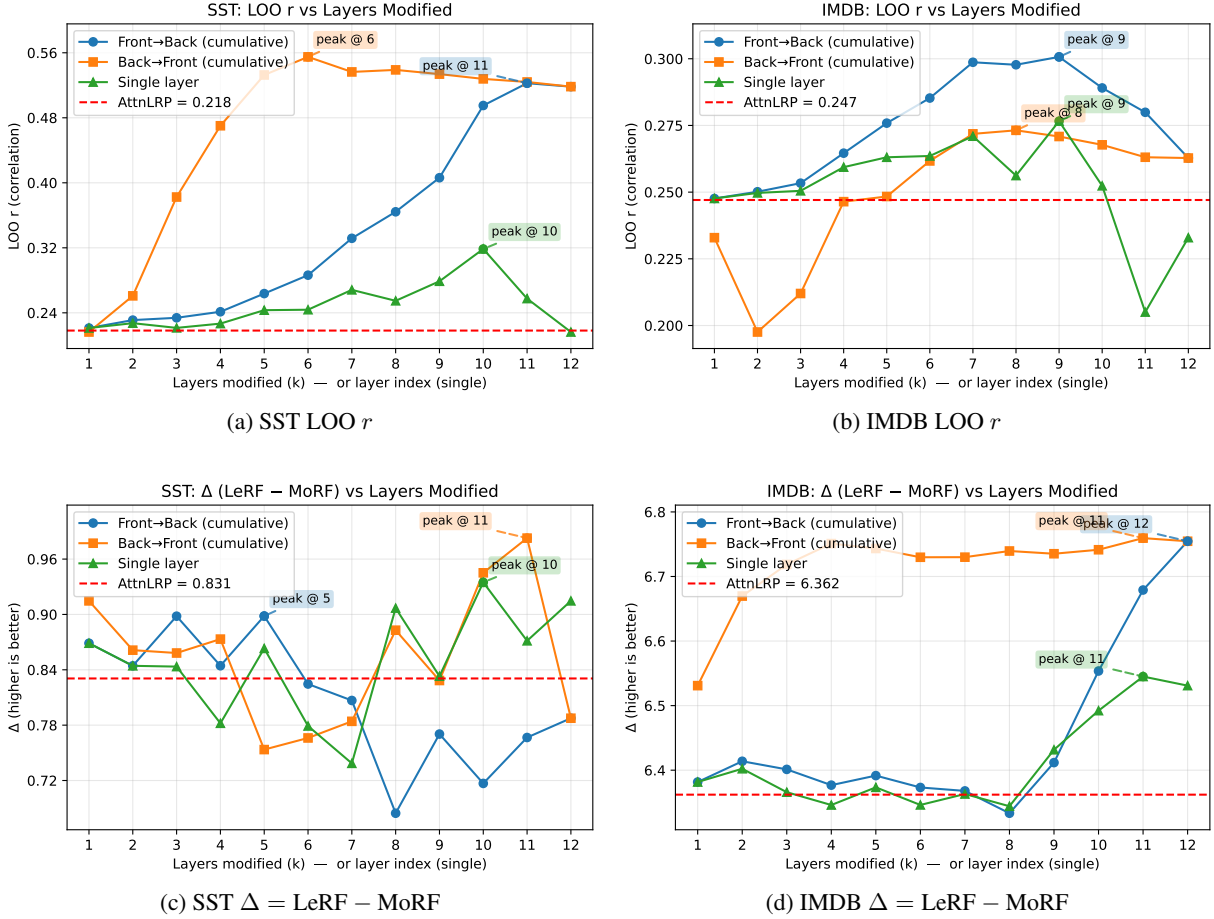


Figure 2: **Layerwise impact of bypassing softmax with CP-LRP on attribution faithfulness.** We compare Pearson correlation with LOO (r , higher is better) and the AOPC difference ($\Delta = \text{LeRF} - \text{MoRF}$, higher is better) on SST and IMDB. We ablate the softmax propagation rule by removing it from: the first k layers (Front-to-Back), the last k layers (Back-to-Front), or only a single layer. **The dashed red line indicates the performance of the standard AttnLRP baseline; points above this line show an improvement in faithfulness.** The largest gains appear in several middle and later layers, consistent across both metrics. For complete numerical results, see Table A3 in the appendix.

5 Related Work

Approximating Feature Importance. Much research has sought efficient approximations for the computationally prohibitive Leave-One-Out (LOO) method. Early heuristics such as Attention Rollout (Abnar and Zuidema, 2020) offered intuitive ways to trace information flow through attention layers, but attention weights themselves have been challenged as faithful explanations (Jain and Wallace, 2019), with follow-up work clarifying conditions under which they can be informative (Wiegreffe and Pinter, 2019). More principled approaches based on relevance conservation, most notably Layer-Wise Relevance Propagation (LRP) (Bach et al., 2015), provide efficient single-pass alternatives, spurring adaptations for Transformers (Ali et al., 2022) and extensions to bilinear settings and

attention mechanisms, including BiLRP for dot-product similarity models (Eberle et al., 2022) and AttnLRP for Transformers (Achtibat et al., 2024). These methods focus on deriving propagation rules and have shown promise as efficient substitutes for LOO, but their axiomatic properties remain under-explored—especially in modern architectures with complex bilinear interactions.

Implementation Invariance and Canonization. The reliability of attribution methods is often judged by formal axioms (Sundararajan et al., 2017), among which *implementation invariance* plays a central role: explanations should depend only on a model’s function, not its specific parameterization (Kindermans et al., 2022). While methods like Integrated Gradients (IG) satisfy this axiom by design, propagation-based methods such

as LRP (Montavon et al., 2018; Shrikumar et al., 2017) do not. Sundararajan et al. (2017) first illustrated this flaw by rearranging ReLU nonlinearities in functionally equivalent networks, leading to different LRP attributions. Subsequent works find similar violations in BatchNorm layers and solve it with *model canonization*: merging BatchNorm with preceding convolutions to stabilize explanations and reduce implementation-dependent artifacts (Guillemot et al., 2020; Yeom et al., 2021). While effective for CNNs, these techniques do not address the core propagation rules for bilinear and softmax layers in Transformers, which rely on LayerNorm instead of BatchNorm.

Perspectives on Bilinear Layers. Bilinear layers, a core component of attention, have been studied from both axiomatic and mechanistic perspectives. From an axiomatic standpoint, recent theoretical results show that no attribution method assigning relevance to individual features can faithfully explain polynomial functions with correlated inputs, motivating group-based attributions in bilinear settings (You et al., 2025a). Mechanistic interpretability work treats bilinear operations as the structural backbone of “attention circuits,” enabling reverse-engineering of model computations (Elhage et al., 2021; Nanda et al., 2023). Bilinear layers have also been shown to admit linear tensor decompositions that expose pairwise interactions, making them mathematically tractable for analysis (Sharkey, 2023). By contrast, our work focuses on *implementation invariance*, formally proving that LRP’s propagation rules for bilinear layers violate this axiom even in simplified settings.

Evaluating Explanations. Evaluating explanations involves multiple desiderata that capture different aspects of explanatory quality, since no single metric suffices (Jacovi et al., 2021; Atanasova et al., 2023). Key desiderata include *faithfulness*, *stability and consistency*, *structural properties*, and *expert alignment*. *Faithfulness* assesses how well explanations reflect the model’s behavior. Standard approaches include sanity checks (Adebayo et al., 2018), retraining-based benchmarks (Hooker et al., 2018), and post-hoc metrics such as Leave-One-Out (LOO) and insertion/deletion curves (Lundberg and Lee, 2017; Petsiuk et al., 2018; Samek et al., 2017; Atanasova et al., 2023; Feng et al., 2018). *Stability* assesses the robustness of explanations to perturbations in explanations or inputs (Slack et al., 2021; Xue et al., 2023; Kim et al.,

2024; Jin et al., 2025b; You et al., 2025b). *Structural properties*, such as contiguity or sparsity, evaluate whether explanations form coherent and interpretable patterns rather than fragmented noise (Kim et al., 2024; You et al., 2025a). Beyond these, other work has proposed causal or environment-invariant criteria for explanations, aiming to identify rationales that remain predictive across different environments (Chang et al., 2020). *Expert alignment* measures agreement with human or domain-expert expectations (Doshi-Velez and Kim, 2017; Jin et al., 2025a; Havaldar et al., 2025; Lage et al., 2019; Nguyen, 2018). In this work, we focus on efficiently approximating LOO and complement it with standard perturbation-based metrics.

6 Conclusion

We examine the conditions under which LRP-style attributions align with Leave-One-Out (LOO) in Transformers. We identify two structural sources of mismatch: (1) bilinear rules in AttnLRP are not implementation invariant, and (2) softmax propagation introduces structural and linearization errors that distort relevance. We empirically show that (1) in linear attention, AttnLRP’s bilinear rules indeed lead to implementation-variant attributions, and that (2) propagating only through value matrices while bypassing softmax in attention layers, as in CP-LRP, leads to better correlation with LOO. We hope this work motivates future work on efficient LOO attribution approximation methods with explicit theoretical guarantees.

Limitations

This study is limited to a toy analytic example, BERT, and a simple linear attention network; generalization to other attention-based architectures remains to be explored. Our analysis focuses on attention layers under a specific set of LRP design choices, and does not exhaustively compare alternative propagation rules or gradient-based attribution methods. Our LOO reference is based on masking-based removal, which may induce distribution shift. Evaluation primarily relies on Pearson correlation with LOO and perturbation-based metrics, though other evaluation metrics exist. Finally, while we identify several failure modes of current LRP variants, we do not propose a remedy; developing axiomatically grounded and efficient alternatives is left for future work.

Acknowledgments

Part of this work was done when Weiqiu You and Siqi Zeng were visiting research students at Okinawa Institute of Science and Technology. WY was supported by a gift from AWS AI to Penn Engineering’s ASSET Center for Trustworthy AI. SZ and HZ were partly supported by an NSF IIS grant No. 2416897 and an NSF CAREER Award No. 2442290. HZ would like to thank the support of a Google Research Scholar Award and Nvidia Academic Grant Award. MY was partly supported by JSPS KAKENHI Grant Number 24K03004 and by JST ASPIRE JPMJAP2302. The views and conclusions expressed in this paper are solely those of the authors and do not necessarily reflect the official policies or positions of the supporting companies and government agencies.

References

- Samira Abnar and Willem Zuidema. 2020. [Quantifying attention flow in transformers](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4190–4197, Online. Association for Computational Linguistics.
- Reduan Achitbat, Sayed Mohammad Vakilzadeh Hatefi, Maximilian Dreyer, Aakriti Jain, Thomas Wiegand, Sebastian Lapuschkin, and Wojciech Samek. 2024. [AttnLRP: Attention-aware layer-wise relevance propagation for transformers](#). In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 135–168. PMLR.
- Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. 2018. [Sanity checks for saliency maps](#). In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.
- Ameen Ali, Thomas Schnake, Oliver Eberle, Grégoire Montavon, Klaus-Robert Müller, and Lior Wolf. 2022. [XAI for transformers: Better explanations through conservative propagation](#). In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 435–451. PMLR.
- Marco Ancona, Enea Ceolini, Cengiz Öztireli, and Markus Gross. 2018. [Towards better understanding of gradient-based attribution methods for deep neural networks](#). In *International Conference on Learning Representations*.
- Pepa Atanasova, Oana-Maria Camburu, Christina Lioma, Thomas Lukasiewicz, Jakob Grue Simonsen, and Isabelle Augenstein. 2023. [Faithfulness tests for natural language explanations](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 283–294, Toronto, Canada. Association for Computational Linguistics.
- Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. 2015. [On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation](#). *PloS one*, 10(7):e0130140.
- Shiyu Chang, Yang Zhang, Mo Yu, and Tommi Jaakkola. 2020. [Invariant rationalization](#). In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 1448–1458. PMLR.
- Ian Covert, Scott Lundberg, and Su-In Lee. 2021. [Explaining by removing: A unified framework for model explanation](#). *Journal of Machine Learning Research*, 22(209):1–90.
- 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.
- Finale Doshi-Velez and Been Kim. 2017. [Towards a rigorous science of interpretable machine learning](#). *Preprint*, arXiv:1702.08608.
- Oliver Eberle, Jochen Buttner, Florian Krautli, Klaus-Robert Müller, Matteo Valleriani, and Grégoire Montavon. 2022. [Building and Interpreting Deep Similarity Models](#). *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 44(03):1149–1161.
- Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, and 6 others. 2021. [A mathematical framework for transformer circuits](#). *Transformer Circuits Thread*.
- Shi Feng, Eric Wallace, Alvin Grissom II, Mohit Iyyer, Pedro Rodriguez, and Jordan Boyd-Graber. 2018. [Pathologies of neural models make interpretations difficult](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3719–3728, Brussels, Belgium. Association for Computational Linguistics.
- Mathilde Guillemot, Catherine Heusele, Rodolphe Korch, Sylvianne Schnebert, and Liming Chen. 2020. [Breaking batch normalization for better explainability of deep neural networks through layer-wise relevance propagation](#). *Preprint*, arXiv:2002.11018.

- Mohamed Saif Hameed, Simon Laplante, Caterina Masino, Muhammad Khalid, Haochi Zhang, Sergey Protserov, Jaryd Hunter, Pouria Mashouri, Andras Fecso, Michael Brudno, and Amin Madani. 2023. [What is the educational value and clinical utility of artificial intelligence for intraoperative and postoperative video analysis? a survey of surgeons and trainees.](#) *Surgical Endoscopy*, 37.
- Shreya Havaldar, Helen Jin, Chaehyeon Kim, Anton Xue, Weiqiu You, Gary E. Weissman, Rajat Deo, Sameed Khatana, Helen Qu, Marco Gatti, Daniel A. Hashimoto, Amin Madani, Masao Sako, Bhuvnesh Jain, Lyle Ungar, and Eric Wong. 2025. [T-fix: Text-based explanations with features interpretable to experts.](#)
- Sara Hooker, D. Erhan, Pieter-Jan Kindermans, and Been Kim. 2018. [A benchmark for interpretability methods in deep neural networks.](#) In *Neural Information Processing Systems*.
- Alon Jacovi, Ana Marasović, Tim Miller, and Yoav Goldberg. 2021. [Formalizing trust in artificial intelligence: Prerequisites, causes and goals of human trust in ai.](#) In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency, FAccT '21*, page 624–635, New York, NY, USA. Association for Computing Machinery.
- Sarthak Jain and Byron C. Wallace. 2019. [Attention is not Explanation.](#) 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 3543–3556, Minneapolis, Minnesota. Association for Computational Linguistics.
- Helen Jin, Shreya Havaldar, Chaehyeon Kim, Anton Xue, Weiqiu You, Helen Qu, Marco Gatti, Daniel A. Hashimoto, Bhuvnesh Jain, Amin Madani, Masao Sako, Lyle Ungar, and Eric Wong. 2025a. [The FIX benchmark: Extracting features interpretable to experts.](#) *Journal of Data-centric Machine Learning Research*.
- Helen Jin, Anton Xue, Weiqiu You, Surbhi Goel, and Eric Wong. 2025b. [Probabilistic stability guarantees for feature attributions.](#) In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.
- Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. 2020. [Transformers are RNNs: Fast autoregressive transformers with linear attention.](#) In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 5156–5165. PMLR.
- Chaehyeon Kim, Weiqiu You, Shreya Havaldar, and Eric Wong. 2024. [Evaluating groups of features via consistency, contiguity, and stability.](#) In *The Second Tiny Papers Track at ICLR 2024*.
- Pieter-Jan Kindermans, Sara Hooker, Julius Adembayo, Maximilian Alber, Kristof T. Schütt, Sven Dähne, Dumitru Erhan, and Been Kim. 2022. [The \(Un\)reliability of Saliency Methods](#), page 267–280. Springer-Verlag, Berlin, Heidelberg.
- Isaac Lage, Emily Chen, Jeffrey He, Menaka Narayanan, Been Kim, Samuel J. Gershman, and Finale Doshi-Velez. 2019. [Human evaluation of models built for interpretability.](#) *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, 7(1):59–67.
- Yann LeCun. 1998. [The mnist database of handwritten digits.](#) <http://yann.lecun.com/exdb/mnist/>.
- Scott M. Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS'17*, page 4768–4777, Red Hook, NY, USA. Curran Associates Inc.
- Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. [Learning word vectors for sentiment analysis.](#) In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.
- Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. 2017. [Explaining nonlinear classification decisions with deep taylor decomposition.](#) *Pattern Recognition*, 65:211–222.
- Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. 2018. [Methods for interpreting and understanding deep neural networks.](#) *Digital Signal Processing*, 73:1–15.
- Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. 2023. [Progress measures for grokking via mechanistic interpretability.](#) In *The Eleventh International Conference on Learning Representations*.
- Dong Nguyen. 2018. [Comparing automatic and human evaluation of local explanations for text classification.](#) In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1069–1078, New Orleans, Louisiana. Association for Computational Linguistics.
- Vitali Petsiuk, Abir Das, and Kate Saenko. 2018. [Rise: Randomized input sampling for explanation of black-box models.](#) In *British Machine Vision Conference*.
- Wojciech Samek, Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, and Klaus-Robert Müller. 2017. [Evaluating the visualization of what a deep neural network has learned.](#) *IEEE Transactions*

- on *Neural Networks and Learning Systems*, 28:2660–2673.
- Lee Sharkey. 2023. [A technical note on bilinear layers for interpretability](#). *Preprint*, arXiv:2305.03452.
- Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. 2017. Learning important features through propagating activation differences. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70, ICML’17*, page 3145–3153. JMLR.org.
- Dylan Z Slack, Sophie Hilgard, Himabindu Lakkaraju, and Sameer Singh. 2021. [Counterfactual explanations can be manipulated](#). In *Advances in Neural Information Processing Systems*.
- Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. [Recursive deep models for semantic compositionality over a sentiment treebank](#). In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.
- Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70, ICML’17*, page 3319–3328. JMLR.org.
- Erico Tjoa and Cuntai Guan. 2019. [A survey on explainable artificial intelligence \(XAI\): towards medical XAI](#). *CoRR*, abs/1907.07374.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. [Attention is all you need](#). In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.
- Rebecca Wexler. 2017. [When a computer program keeps you in jail: How computers are harming criminal justice](#). *The New York Times*. Opinion.
- Sarah Wiegrefe and Yuval Pinter. 2019. [Attention is not not explanation](#). 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 11–20, Hong Kong, China. Association for Computational Linguistics.
- Anton Xue, Rajeev Alur, and Eric Wong. 2023. [Stability guarantees for feature attributions with multiplicative smoothing](#). In *Thirty-seventh Conference on Neural Information Processing Systems*.
- Seul-Ki Yeom, Philipp Seegerer, Sebastian Lapuschkin, Alexander Binder, Simon Wiedemann, Klaus-Robert Müller, and Wojciech Samek. 2021. [Pruning by explaining: A novel criterion for deep neural network pruning](#). *Pattern Recognition*, 115:107899.
- Weiyou You, Helen Qu, Marco Gatti, Bhuvnesh Jain, and Eric Wong. 2025a. [Sum-of-parts: Self-attributing neural networks with end-to-end learning of feature groups](#). In *Forty-second International Conference on Machine Learning*.
- Weiyou You, Simeng Sun, and Mohit Iyyer. 2020. [Hard-coded Gaussian attention for neural machine translation](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7689–7700, Online. Association for Computational Linguistics.
- Weiyou You, Anton Xue, Shreya Havaldar, Delip Rao, Helen Jin, Chris Callison-Burch, and Eric Wong. 2025b. [Probabilistic soundness guarantees in LLM reasoning chains](#). In *The 2025 Conference on Empirical Methods in Natural Language Processing*.
- Jiaming Zeng, Berk Ustun, and Cynthia Rudin. 2016. [Interpretable classification models for recidivism prediction](#). *Journal of the Royal Statistical Society Series A: Statistics in Society*, 180(3):689–722.

A Comparison with Integrated Gradients

Integrated Gradients (IG) is an axiomatically-sound method that satisfies implementation invariance by design (Sundararajan et al., 2017). It computes feature attributions by integrating gradients along a path from a baseline input x' to the actual input x :

$$\text{IG}_i(x) = (x_i - x'_i) \int_0^1 \frac{\partial f(x' + \alpha(x - x'))}{\partial x_i} d\alpha$$

Because IG’s attribution depends only on the function f and not the network’s specific architecture, it would produce identical explanations for both the scalar and attention counterexamples discussed in Section 3, regardless of how the operations are grouped. While IG is computationally more intensive, this contrast highlights a fundamental trade-off: propagation-based methods like LRP offer efficiency but can fail to preserve fundamental axioms.

B Experiment Details

B.1 Hyperparameters

B.1.1 MNIST Linear Attention Experiments

For MNIST, we train simple models QKVLeftAssoc_Seq_NoSoftmax and QKVRightAssoc_Seq_NoSoftmax, which are linear attention networks without softmax normalization. Input images are resized from 28×28 to 14×14 and flattened into sequences of length 196. Both models use an embedding dimension of 32 and predict over 10 classes. We train each

Table A3: Layer-wise effects for bypassing softmax in backpropagation on attribution metrics (SST top, IMDB bottom). Metrics: LOO r , LeRF, MoRF, and $\Delta = \text{LeRF} - \text{MoRF}$. $L_{:3}$ means removing layers 1-3. $L_{7:}$ means removing layers 7-12.

Front $\rightarrow$ Back					Back $\rightarrow$ Front					Single Layer				
Rm Layers	LOO r	LeRF	MoRF	Δ	Rm Layers	LOO r	LeRF	MoRF	Δ	Rm Layer	LOO r	LeRF	MoRF	Δ
SST														
$L_{:1}$	0.22	0.92	0.05	0.87	$L_{12:}$	0.22	0.89	-0.02	0.91	L_1	0.22	0.92	0.05	0.87
$L_{:2}$	0.23	0.89	0.05	0.84	$L_{11:}$	0.26	0.91	0.05	0.86	L_2	0.23	0.89	0.05	0.84
$L_{:3}$	0.23	0.89	-0.01	0.90	$L_{10:}$	0.38	0.90	0.04	0.86	L_3	0.22	0.88	0.04	0.84
$L_{:4}$	0.24	0.90	0.05	0.84	$L_{9:}$	0.47	0.97	0.10	0.87	L_4	0.23	0.84	0.06	0.78
$L_{:5}$	0.26	0.88	-0.02	0.90	$L_{8:}$	0.53	0.93	0.18	0.75	L_5	0.24	0.91	0.05	0.86
$L_{:6}$	0.29	0.88	0.06	0.82	$L_{7:}$	0.55	0.97	0.20	0.77	L_6	0.24	0.90	0.12	0.78
$L_{:7}$	0.33	0.82	0.02	0.81	$L_{6:}$	0.54	0.96	0.18	0.78	L_7	0.27	0.89	0.15	0.74
$L_{:8}$	0.36	0.82	0.14	0.68	$L_{5:}$	0.54	1.06	0.17	0.88	L_8	0.25	0.96	0.05	0.91
$L_{:9}$	0.41	0.91	0.14	0.77	$L_{4:}$	0.53	1.01	0.18	0.83	L_9	0.28	0.93	0.09	0.83
$L_{:10}$	0.50	0.85	0.14	0.72	$L_{3:}$	0.53	1.11	0.16	0.95	L_{10}	0.32	0.93	-0.01	0.93
$L_{:11}$	0.52	0.94	0.18	0.77	$L_{2:}$	0.52	1.12	0.13	0.98	L_{11}	0.26	0.91	0.04	0.87
$L_{:12}$	0.52	1.00	0.22	0.79	$L_{1:}$	0.52	1.00	0.22	0.79	L_{12}	0.22	0.89	-0.02	0.91
IMDB														
$L_{:1}$	0.25	3.72	-2.66	6.38	$L_{12:}$	0.23	3.72	-2.81	6.53	L_1	0.25	3.72	-2.66	6.38
$L_{:2}$	0.25	3.72	-2.69	6.41	$L_{11:}$	0.20	3.72	-2.95	6.67	L_2	0.25	3.72	-2.68	6.40
$L_{:3}$	0.25	3.73	-2.68	6.40	$L_{10:}$	0.21	3.72	-3.00	6.72	L_3	0.25	3.72	-2.64	6.37
$L_{:4}$	0.26	3.73	-2.65	6.38	$L_{9:}$	0.25	3.73	-3.02	6.75	L_4	0.26	3.72	-2.63	6.35
$L_{:5}$	0.28	3.73	-2.67	6.39	$L_{8:}$	0.25	3.72	-3.02	6.74	L_5	0.26	3.72	-2.65	6.37
$L_{:6}$	0.29	3.72	-2.65	6.37	$L_{7:}$	0.26	3.72	-3.01	6.73	L_6	0.26	3.72	-2.63	6.35
$L_{:7}$	0.30	3.73	-2.64	6.37	$L_{6:}$	0.27	3.73	-3.00	6.73	L_7	0.27	3.72	-2.64	6.36
$L_{:8}$	0.30	3.72	-2.62	6.33	$L_{5:}$	0.27	3.73	-3.01	6.74	L_8	0.26	3.72	-2.63	6.34
$L_{:9}$	0.30	3.73	-2.69	6.41	$L_{4:}$	0.27	3.73	-3.00	6.74	L_9	0.28	3.72	-2.71	6.43
$L_{:10}$	0.29	3.74	-2.81	6.55	$L_{3:}$	0.27	3.73	-3.01	6.74	L_{10}	0.25	3.73	-2.77	6.49
$L_{:11}$	0.28	3.74	-2.94	6.68	$L_{2:}$	0.26	3.74	-3.02	6.76	L_{11}	0.21	3.72	-2.82	6.55
$L_{:12}$	0.26	3.73	-3.02	6.75	$L_{1:}$	0.26	3.73	-3.02	6.75	L_{12}	0.23	3.72	-2.81	6.53

model for 5 epochs using Adam (learning rate 0.001, $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$) with cross-entropy loss.

For LOO, we remove *single pixels* on the 14×14 resized images by setting them to zero and measuring the change in the predicted class logit (batch size 8 for pixel sweeps). Attribution is computed with LRP- ϵ ($\epsilon = 10^{-6}$), and attributions are normalized to sum to 1 before computing Pearson correlation with LOO, complemented by MoRF/LeRF perturbation curves.

B.1.2 Text Experiments

For text experiments, we use pretrained BERT-base models from Hugging Face. For SST, we use textattack/bert-base-uncased-SST-2, and fabriceyh/bert-base-uncased-imdb for IMDB. Both models are standard BERT-base architectures fine-tuned for sentiment classification.

LOO granularity. For both SST and IMDB, LOO is computed at the **single-token** level by masking one token at a time and measuring the change in the predicted class logit.

MoRF/LeRF granularity. For IMDB only, MoRF and LeRF perturbation curves are computed over **non-overlapping contiguous 16-token chunks** to make perturbation more computationally tractable on long sequences. For SST, MoRF and LeRF use **single-token** removals. This grouping affects only the perturbation curves; all LOO-based correlations use single-token LOO.

B.2 Full Results

Table A3 provides the full numerical results for our layer-wise softmax ablation study, corresponding to the data visualized in Figure 2 in the main text.