TU WIEN Informatics

Faithful Attention Attribution in Vision Transformers for Chest X-Ray Interpretation

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Logic and Computation

by

Julius Šula
Registration Number 11914972

to the Faculty of Informatics

at the TU Wien

Advisor: Univ.Prof. Dipl.-Inf. Dr.rer.nat. Thomas Lukasiewicz
Assistance: Dr. Bayar Ilhan Menzat

Vienna, January 1, 2026

Julius Šula	Thomas Lukasiewicz

Erklärung zur Verfassung der Arbeit

Julius Šula

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.
Ich erkläre weiters, dass ich mich generativer KI-Tools lediglich als Hilfsmittel bedient habe und in der vorliegenden Arbeit mein gestalterischer Einfluss überwiegt. Im Anhang „Übersicht verwendeter Hilfsmittel“ habe ich alle generativen KI-Tools gelistet, die verwendet wurden, und angegeben, wo und wie sie verwendet wurden. Für Textpassagen, die ohne substantielle Änderungen übernommen wurden, haben ich jeweils die von mir formulierten Eingaben (Prompts) und die verwendete IT- Anwendung mit ihrem Produktnamen und Versionsnummer/Datum angegeben.

Wien, 1. Jänner 2026

Julius Šula

Acknowledgements

Abstract

Vision Transformers (ViTs) achieve strong performance in natural and medical imaging, yet their decision processes remain opaque—especially problematic in high-stakes settings like chest X-ray interpretation. TransMM is among the strongest attribution methods for ViTs, combining attention with class-specific gradients to highlight influential image patches. We ask whether injecting *semantic structure* from Sparse Autoencoders (SAEs) can further improve the faithfulness of such attributions.

We introduce *Feature-Gradient Attribution*, which extends TransMM's principle from attention space to *feature* space. SAEs are trained on residual streams to decompose activations into sparse, interpretable features, providing per-patch feature activations. We project gradients onto the SAE feature basis and compute feature-gradient scores that capture both *which* learned features are present and *how* they influence the target logit. These scores yield per-patch gates that modulate TransMM's attention maps before relevance propagation, forming a lightweight, semantically informed correction.

Across three datasets (chest X-rays, endoscopy, natural images), two architectures (fine-tuned ViT-B/16 and contrastively pre-trained CLIP ViT-B/32), and three complementary faithfulness metrics, our method improves attribution faithfulness consistently. Improvements are statistically significant ($p<0.001$) on all three metrics for one dataset and on two of three metrics for the remaining datasets. We observe gains of 10.5–34.8% on SaCo and 9.7–43.0% on Faithfulness Correlation, with Pixel Flipping improving by 1.8–10.8%. Notably, we never observe degradation relative to TransMM on any metric–dataset combination.

Contents

CHAPTER 1

Introduction

Vision Transformers [DBK+21] have become increasingly prominent in computer vision, demonstrating competitive or superior performance to convolutional networks across diverse tasks when sufficient training data is available. Their self-attention mechanisms enable modeling of global dependencies, which has proven valuable in domains ranging from natural image classification to medical imaging applications such as chest X-ray diagnosis [MBSP21, KBB23] and gastrointestinal endoscopy [SM24]. However, like other deep neural networks, Vision Transformers' decision-making processes remain largely opaque, creating a critical challenge for deployment in high-stakes applications where understanding *why* a model makes a prediction is as important as the prediction itself.

1.1 The Need for Faithful Explanations in Medical AI

In medical imaging, the stakes of model deployment extend beyond predictive accuracy. Clinicians must understand which image regions drive diagnostic predictions to verify that models have learned clinically meaningful patterns rather than spurious correlations. Research on clinical decision-making demonstrates that multimodal explanations can significantly improve physicians' ability to detect model failures and biases [KME+24]. A model that achieves high accuracy by exploiting dataset biases, such as hospital-specific imaging artifacts or irrelevant anatomical markers for instance, poses serious risks in clinical practice. Improving the faithfulness of visual attribution maps therefore directly impacts the quality of explanations that clinicians can use to validate model reasoning alongside other interpretability modalities.

TransMM (Transformer Attribution using Multimodal Mixing) [CGW21b] has established itself as a leading attribution method for Vision Transformers [WKT+24a, KBB23, AHD+24]. Its elegance lies in a simple principle: combine attention weights with gradients to produce class-specific attribution maps. By weighting attention maps with the gradient of the target class with respect to those attention weights, TransMM identifies which

patches influence predictions while accounting for the complex information flow through transformer architectures.

1.2 From Attention to Semantic Features

Sparse Autoencoders as interpretability tools. Recent advances in mechanistic interpretability have introduced Sparse Autoencoders (SAEs) as a powerful tool for understanding what models learn [BTB+23, HCS+24]. SAEs address a fundamental challenge in neural network interpretability: individual neurons typically respond to multiple unrelated concepts (polysemanticity), making it difficult to understand what the network has learned. By learning overcomplete sparse representations, SAEs decompose these entangled activations into interpretable features where each dimension corresponds to a distinct semantic concept. These concepts range from low-level textures to high-level objects and abstract patterns. Recent work has demonstrated their effectiveness in vision domains, confirming that SAE features capture genuine computational primitives that causally influence model behavior [JSH+25, Ano25, LCCS25].

The attribution-interpretability gap. While attribution methods and mechanistic interpretability have both advanced significantly, they have largely evolved in parallel, creating a gap between explaining *where* models attend and understanding *what semantic concepts* drive those decisions. Recent work has begun exploring connections: using attribution to explain SAE features [HKK25], or using SAE features to validate attributions post-hoc [SCBWS25]. However, neither approach integrates semantic feature information directly into the attribution computation itself. If SAE features capture meaningful semantic concepts and attribution methods aim to identify important image regions, can we leverage these learned representations to enhance attribution quality during computation rather than after?

Extending TransMM's principle to feature space. This thesis investigates whether TransMM's core principle of combining activation patterns with gradient information can be extended from attention maps to semantic feature space, yielding more faithful attributions. TransMM's success stems from recognizing that attention weights alone are insufficient: knowing where the model attends (attention patterns) must be combined with knowing how that attention affects predictions (gradients). We propose that a parallel limitation exists for gradients themselves: raw gradients aggregate influences across all learned representations, potentially conflating relevant semantic features with noise. Our approach extends TransMM's principle to the feature level. For each image patch, we extract sparse feature activations from the model's residual stream using SAEs and compute gradients with respect to these features. The product of feature activation (indicating which semantic concepts are present) and feature gradient (indicating how those concepts influence the prediction) provides a semantic importance score. Features that are both active in a patch and gradient-aligned with the target prediction receive high scores, while features that are absent or irrelevant receive low scores. We aggregate these feature-gradient scores and use them to modulate TransMM's attention maps before

relevancy propagation, creating a correction mechanism that respects the model's learned semantic structure. The method maintains TransMM's architectural simplicity, requiring only SAE inference and element-wise operations, while adding minimal computational overhead.

Through comprehensive evaluation across three diverse datasets, two model architectures, and three complementary faithfulness metrics, we demonstrate consistent improvements over TransMM, with gains of 10–43% depending on metric and dataset. This consistency across diverse experimental conditions provides strong evidence that incorporating semantic structure through SAE features represents a genuine advancement in attribution quality for Vision Transformers.

1.3 Contributions

This thesis makes the following contributions to attribution methods for Vision Transformers:

1. **Novel integration of mechanistic interpretability with attribution**: We present a method that incorporates SAE feature-gradient decomposition directly into attribution computation, modulating attention maps before relevancy propagation. Unlike prior work that uses attribution to explain features [HKK25] or features to validate attribution [SCBWS25], we demonstrate that semantic features can enhance attribution faithfulness when integrated into the computation itself.

2. **Principled feature-gradient decomposition**: We develop a method to decompose residual stream gradients through SAE feature space, identifying which semantic features drive patch importance. This decomposition provides interpretable per-patch scores that reflect both feature presence and gradient alignment, extending TransMM's attention-gradient principle to the feature level.

3. **Practical integration with state-of-the-art attribution**: We show how to incorporate feature-gradient signals into TransMM through attention map modulation, preserving its computational efficiency and architectural simplicity while leveraging semantic structure from mechanistic interpretability.

4. **Comprehensive empirical validation**: We demonstrate substantial improvements across three datasets (two medical imaging tasks and natural images), two model architectures (fine-tuned and contrastively pre-trained), and three complementary faithfulness metrics, with improvements of 10-43% over the TransMM baseline.

5. **Methodological insights**: Through systematic ablation studies, we establish that attribution improvement requires the interaction between feature activations and gradients, neither signal alone suffices. We also identify optimal layer ranges and demonstrate that multi-layer feature combinations provide synergistic benefits.

1.4 Research Questions

This thesis addresses the following research questions:

1. **Can SAE features improve attribution faithfulness in Vision Transformers?** We investigate whether incorporating semantic features into gradient-based attribution yields measurable improvements across established faithfulness metrics including correlation, magnitude alignment, and ranking quality.

2. **What is the optimal way to combine feature and gradient information?** Through ablation studies, we test whether attribution improvement stems from feature activations, decomposed gradients, or specifically their interaction, determining which signal components are necessary for faithful attribution.

3. **Does the approach generalize across domains and architectures?** We evaluate robustness across medical imaging (chest X-rays, endoscopy) and natural images, fine-tuned and contrastively pre-trained models, and different Vision Transformer architectures to assess broad applicability.

1.5 Thesis Structure

The remainder of this thesis is organized as follows:

Chapter 2 provides background on Vision Transformers, covering their architecture and self-attention mechanisms that make attribution challenging, then reviews attribution methods for Vision Transformers, tracing the evolution from Layer-wise Relevance Propagation through TransMM and establishing the state-of-the-art baseline for our work.

Chapter 3 presents mechanistic interpretability and Sparse Autoencoders, explaining how SAEs decompose polysemantic neurons into interpretable features and reviewing recent applications to vision models, then introduces faithfulness metrics for evaluating attribution quality, discussing the complementary aspects captured by different evaluation protocols and justifying our choice of metrics.

Chapter 4 details our Feature-Gradient Attribution method, including the mathematical framework, gate construction, integration with TransMM, and implementation considerations.

Chapter 5 presents comprehensive experimental evaluation across three datasets, including validation experiments to identify optimal configurations and test set results demonstrating substantial faithfulness improvements.

Chapter 6 discusses implications, limitations, and future directions, concluding with the broader impact on trustworthy AI for medical imaging.

CHAPTER 2

Vision Transformers and Attribution Methods

Vision Transformers [DBK+21] have become increasingly prominent in computer vision, yet their decision-making processes remain not fully explained. This chapter first introduces the Vision Transformer architecture and its unique challenges for interpretability (Section 2.2), then reviews the evolution of attribution methods from Layer-wise Relevance Propagation to TransMM (Section 2.3).

2.1 The Transformer Architecture

The Transformer architecture [VSP+17] represented a turning point in deep learning, addressing fundamental limitations of recurrent neural networks. Unlike RNNs, which process sequences sequentially and suffer from vanishing gradients over long distances [KK01], Transformers enable fully parallel computation through their attention mechanism, allowing efficient training on modern hardware and effective modeling of long-range dependencies.

Originally introduced for neural machine translation, the Transformer's self-attention mechanism has proven remarkably versatile, extending to natural language understanding [DCLT19], protein structure prediction [JEP+21], and computer vision [DBK+21]. Understanding the Transformer's components is essential for analyzing how attribution methods must account for its unique information flow patterns, particularly the global receptive field enabled by self-attention.

2.1.1 Core Components

Encoder and decoder architecture. The transformer consists of two primary components. The encoder processes input sequences into continuous representations capturing

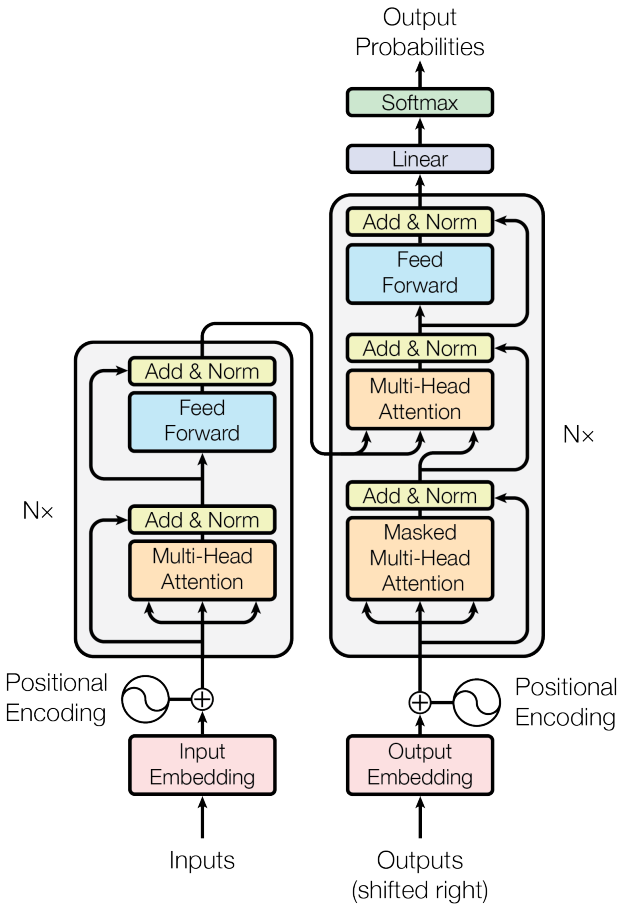


Figure 2.1: The Transformer architecture with encoder (left) and decoder (right) stacks. Each encoder layer contains multi-head self-attention and feed-forward networks, while decoder layers add cross-attention to encoder outputs. Residual connections and layer normalization are applied throughout. Figure adapted from [VSP+17].

semantic and contextual information through N identical layers, each containing multi-head self-attention and position-wise feed-forward networks with residual connections [HZRS16] and layer normalization [BKH16]. The decoder generates output sequences autoregressively with three sub-layers: masked self-attention, encoder-decoder attention, and feed-forward networks. In practice, many applications use only the encoder (e.g., BERT [DCLT19]) or only the decoder (e.g., GPT [RNSS18]). Vision Transformers, which form the basis of this thesis, employ only the encoder architecture.

Attention mechanism. The attention mechanism enables the model to weigh the importance of different input parts when processing each position. Given an input representation, the mechanism computes three vectors for each token: a query $\mathbf{Q}$ (what the token is looking for), a key $\mathbf{K}$ (what the token offers), and a value $\mathbf{V}$ (the actual information to be aggregated).

The core attention operation computes a weighted sum of values, where weights are determined by compatibility between queries and keys:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}}\right)\mathbf{V} \tag{2.1}$$

where d_k is the key dimension. The scaling factor $1/\sqrt{d_k}$ prevents dot products from growing large, which would push softmax into regions with extremely small gradients. The softmax converts compatibility scores into a probability distribution, ensuring attention weights sum to one.

Multi-head attention projects queries, keys, and values h times with different learned projections, allowing the model to attend to information from different representation subspaces simultaneously:

$$\text{MultiHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)\mathbf{W}^O \tag{2.2}$$

$$\text{where head}_i = \text{Attention}(\mathbf{Q}\mathbf{W}_i^Q, \mathbf{K}\mathbf{W}_i^K, \mathbf{V}\mathbf{W}_i^V) \tag{2.3}$$

The original architecture uses $h = 8$ parallel attention heads with $d_k = d_v = d_{\text{model}}/h = 64$. This enables capturing different relationship types simultaneously, for instance, one head might focus on syntactic dependencies while another captures semantic similarities.

Feed-forward networks and positional encoding. Each transformer layer contains a position-wise feed-forward network applied identically to each position: $\text{FFN}(\mathbf{x}) = \max(0, \mathbf{x}\mathbf{W}_1 + \mathbf{b}_1)\mathbf{W}_2 + \mathbf{b}_2$. This consists of two linear transformations with ReLU activation. While attention determines *where* to look, feed-forward networks process *what* has been aggregated. The inner layer typically has dimensionality $d_{ff} = 2048$, four times larger than the model dimension $d_{\text{model}} = 512$.

Since attention operations are permutation-invariant, transformers explicitly encode positional information by adding positional embeddings to input embeddings: $\mathbf{z}_i = \mathbf{x}_i + \mathbf{p}_i$. In Vision Transformers, these embeddings encode the 2D spatial location of image patches, enabling the model to leverage spatial relationships [DBK+21].

2.2 Vision Transformers

While Transformers achieved remarkable success in natural language processing, their application to computer vision remained limited until Dosovitskiy et al. [DBK+21] demonstrated that a pure transformer architecture could match or exceed convolutional neural networks on image classification. This breakthrough challenged the prevailing assumption that CNN inductive biases of translation equivariance and locality were essential for learning visual representations.

2.2.1 Architecture

Vision Transformers adapt the transformer encoder for image classification through a conceptually elegant approach: treating an image as a sequence of patches. An input image $\mathbf{x} \in \mathbb{R}^{H \times W \times C}$ is reshaped into a sequence of flattened 2D patches $\mathbf{x}_p \in \mathbb{R}^{N \times (P^2 \cdot C)}$, where (H, W) is the original resolution, C is the number of channels (typically 3 for RGB), (P, P) is the patch resolution (commonly 16×16 or 32×32), and $N = HW/P^2$ is the resulting number of patches. For a standard 224×224 image with 16×16 patches, this yields $N = 196$ patch tokens.

Patch embedding and position encoding. Each flattened patch is linearly projected to model dimension D through a trainable embedding matrix $\mathbf{E} \in \mathbb{R}^{(P^2 \cdot C) \times D}$:

$$\mathbf{z}_0 = [\mathbf{x}_{\text{class}}; \mathbf{x}_p^1\mathbf{E}; \mathbf{x}_p^2\mathbf{E}; \ldots; \mathbf{x}_p^N\mathbf{E}] + \mathbf{E}_{\text{pos}} \tag{2.4}$$

where $\mathbf{x}_{\text{class}}$ is a learnable class token prepended to the sequence, serving as an aggregated image representation for classification (analogous to BERT's [CLS] token [DCLT19]). The positional embeddings $\mathbf{E}_{\text{pos}} \in \mathbb{R}^{(N+1) \times D}$ are learned parameters encoding each patch's 2D spatial location.

Transformer encoder and classification. The sequence of embedded patches is processed by L transformer encoder layers (typically $L = 12$ for ViT-Base). Each layer applies multi-head self-attention followed by a feed-forward network, with residual connections and layer normalization:

$$\mathbf{z}'_\ell = \text{MSA}(\text{LN}(\mathbf{z}_{\ell-1})) + \mathbf{z}_{\ell-1} \tag{2.5}$$

$$\mathbf{z}_\ell = \text{MLP}(\text{LN}(\mathbf{z}'_\ell)) + \mathbf{z}'_\ell \tag{2.6}$$

After L layers, the class token's final representation $\mathbf{z}_L^0$ is extracted and passed through a classification head: $\mathbf{y} = \text{softmax}(\mathbf{W}_{\text{head}}\mathbf{z}_L^0)$. Figure 2.2 illustrates the complete architecture.

2.2.2 Scaling Behavior and Comparison to CNNs

Vision Transformers exhibit fundamentally different learning characteristics compared to convolutional neural networks [LBBH98, HZRS16]. A critical finding of Dosovitskiy et al. [DBK+21] is that ViTs require substantially larger training datasets to achieve competitive performance. When trained on moderately sized datasets, ViT-Base underperforms comparable ResNets, which the authors attribute to ViTs lacking the inductive biases that help CNNs generalize from limited data. However, this relationship reverses at scale: when pre-trained on large datasets, Vision Transformers match or exceed state-of-the-art CNNs while requiring less compute to train. This reflects a fundamental trade-off—ViTs must learn visual relationships from data rather than having them encoded architecturally, but this flexibility enables superior performance when sufficient training data is available.

How ViTs differ from CNNs. Raghu et al. [RUK+21] demonstrated that Vision Transformers process visual information fundamentally differently than CNNs. Key

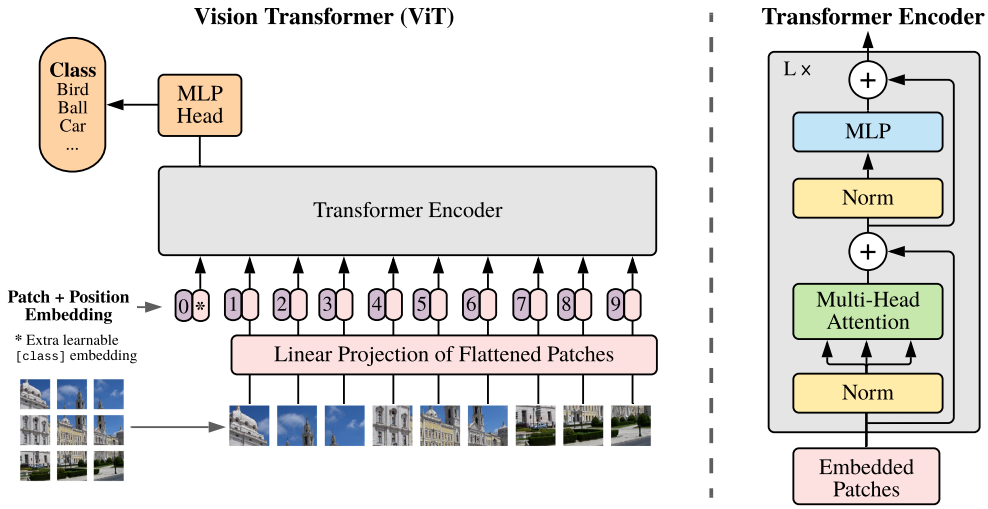


Figure 2.2: Vision Transformer (ViT) architecture. An input image is divided into fixed-size patches, linearly embedded, and augmented with position embeddings. A learnable class token is prepended to the sequence, which is then processed by a standard transformer encoder. The final state of the class token serves as the image representation for classification. Figure adapted from [DBK+21].

distinctions include: uniform representations across layers (contrasting with CNNs' hierarchical progression from edges to objects), early global aggregation through self-attention from the first layer (while CNNs build receptive fields gradually), strong feature propagation through residual connections, and preservation of spatial information throughout the network despite global mixing. Understanding how ViTs aggregate information globally while preserving spatial structure is essential for developing effective explanation methods.

Pre-training paradigms. While the ViT architecture described above is typically trained with supervised classification objectives, alternative pre-training strategies can produce models with different learned representations despite identical architectures. CLIP (Contrastive Language-Image Pre-training) [RKH+21] trains ViT encoders using contrastive learning on image-text pairs, learning to align visual and linguistic representations rather than predict fixed class labels. At inference, CLIP's vision encoder operates identically to a standard ViT, but the contrastive pre-training objective encourages learning features that align with natural language descriptions, potentially creating different semantic structures than supervised fine-tuning [RKH+21]. This thesis evaluates attribution methods on both fine-tuned ViTs (COVID-QU-Ex, Hyperkvasir) and a contrastively pre-trained CLIP model (ImageNet) to assess whether feature-gradient attribution generalizes across these different training paradigms.

Vision Transformers for medical imaging. Vision Transformers have demonstrated strong performance across diverse medical imaging modalities and tasks, including chest

X-ray diagnosis [MBSP21, KBB23], computed tomography analysis [MBSP21], gastrointestinal endoscopy [SM24], and dermatology, pathology, and ophthalmology [CJJ+24], often achieving performance comparable to or exceeding convolutional approaches when sufficient training data is available. While ViTs can achieve strong predictive performance in these domains, their decision-making process remains opaque, creating a critical need for reliable attribution methods that can identify which image regions drive predictions.

2.3 Attribution Methods for Vision Transformers

Vision Transformers have emerged as powerful alternatives to convolutional neural networks [DBK+21], yet their decision-making processes remain opaque. Attribution methods seek to identify which input regions contribute to model predictions, enabling model debugging, verification of learned concepts, and detection of spurious correlations. However, the unique architectural properties of Transformers fundamentally challenge attribution methods developed for convolutional networks. Global receptive fields through self-attention, complex information flow via skip connections, and non-standard activation functions all violate assumptions these methods rely on.

This section traces how attribution methods evolved to address these Transformer-specific challenges, examining why attention weights alone fail as explanations, how Layer-wise Relevance Propagation provides theoretical foundations, and how TransLRP and TransMM extend these principles to Transformers. TransMM serves as the baseline for our work [WKT+24a, KBB23, AHD+24], and understanding its core principle of combining attention patterns with gradient information motivates our investigation of whether this principle extends to sparse semantic features.

2.3.1 The Attribution Challenge in Transformers

The architecture of Vision Transformers presents fundamental challenges for attribution methods originally developed for convolutional networks. Unlike CNNs, which process images through hierarchical local operations, Transformers employ self-attention mechanisms that allow every image patch to interact with every other patch from the first layer [VSP+17]. This global receptive field means that information aggregation is not spatially constrained, making it difficult to trace which input regions influence specific predictions.

Furthermore, Transformers incorporate architectural components that violate the assumptions of traditional attribution methods. Skip connections create multiple information pathways through the network [HZRS16]. LayerNorm operations rescale contributions in complex ways [BKH16]. GELU activation functions produce both positive and negative values [HG17], unlike the strictly non-negative ReLU activations assumed by many attribution techniques. Perhaps most distinctively, Vision Transformers aggregate information through a learned CLS token rather than through spatial pooling operations, fundamentally changing how global information is synthesized for classification [DBK+21].

Why attention weights fail as attribution. The self-attention mechanism naturally produces attention weight matrices that superficially appear to indicate which tokens the model is "attending to" when making predictions. This has led to attention-based attribution methods that interpret these weights directly, from using raw attention weights in single layers to more sophisticated aggregation schemes like attention rollout and attention flow [AZ20].

However, these attention-based methods suffer from fundamental limitations. Jain and Wallace [JW19] demonstrated that attention weights correlate only weakly with gradient-based feature importance measures. They also showed that alternative attention distributions can often yield equivalent predictions, meaning the original attention weights do not provide unique explanations. More problematically, Pruthi et al. [PGD+20] showed that models can be deliberately trained to produce deceptive attention patterns: assigning minimal attention to certain features while continuing to rely heavily on them for predictions. In human studies, these manipulated attention distributions successfully deceived evaluators.

Beyond these empirical findings, attention weights have inherent conceptual limitations. They represent normalized probabilities over tokens, not causal attributions of importance. A token receiving high attention might contribute negatively to the prediction, while a token with low attention might be crucial due to its specific features. Moreover, attention weights only capture the mixing of information in self-attention layers, completely ignoring the substantial computations performed by MLP blocks. Skip connections preserve information outside the attention mechanism, LayerNorm operations rescale contributions, and learned parameters in projection matrices fundamentally transform representations [HZRS16, BKH16]. By focusing solely on attention weights, these methods provide an incomplete and potentially misleading view of how Transformers process information.

2.3.2 Propagation-Based Foundations

To understand modern attribution methods for Transformers, we must first examine the propagation-based techniques from which they evolved. Layer-wise Relevance Propagation and its theoretical connections to Taylor decomposition form the foundation for state-of-the-art Transformer attribution methods like TransLRP and TransMM.

Layer-wise Relevance Propagation. Layer-wise Relevance Propagation (LRP), introduced by Bach et al. [LBM+15], provides a framework for decomposing neural network predictions through backward message-passing. The fundamental distinction between LRP and gradient-based methods lies in the question each addresses. While gradients answer "How would the output change if we perturbed this input?", LRP answers "How much did this input contribute to the actual output?"

LRP operates on the principle of relevance conservation. Starting from the network's output $f(\mathbf{x})$, relevance flows backward through the network layers without being created or destroyed. Formally, if $R_i^{(\ell)}$ denotes the relevance of neuron i in layer ℓ, conservation

requires:

$$\sum_i R_i^{(\ell)} = \sum_j R_j^{(\ell+1)} = \ldots = f(\mathbf{x}) \tag{2.7}$$

This conservation principle ensures that the attribution is complete, meaning every part of the model's decision is accounted for in the final attribution map. The practical implementation requires defining propagation rules that specify how relevance flows between layers. For a standard feedforward layer where neuron j's activation is $a_j = g(\sum_i a_i w_{ij} + b_j)$, the contribution from neuron i to neuron j is the weighted activation $z_{ij} = a_i w_{ij}$. LRP redistributes relevance proportionally to these contributions. The basic LRP rule (LRP-0) is:

$$R_i^{(\ell)} = \sum_j \frac{z_{ij}}{\sum_k z_{kj}} R_j^{(\ell+1)} \tag{2.8}$$

However, this basic rule suffers from numerical instability when $\sum_k z_{kj} \approx 0$. Refined rules address this: the ϵ-rule adds a small stabilizer to prevent division by zero, while the $\alpha\beta$-rule treats positive and negative contributions separately with the constraint $\alpha - \beta = 1$ ensuring conservation.

Deep Taylor Decomposition. Deep Taylor Decomposition (DTD), introduced by Montavon et al. [MLB+17], provides theoretical foundations that connect LRP's empirically successful propagation rules to fundamental mathematical principles. Rather than replacing LRP, DTD demonstrates that these rules can be derived from first principles using Taylor expansions.

The key insight is distinguishing between sensitivity and attribution. Standard backpropagation computes the gradient $\nabla_{\mathbf{x}} f$ at the input point, measuring sensitivity to infinitesimal changes. In contrast, attribution asks: "How much did each input feature contribute to the actual output value $f(\mathbf{x})$?"

DTD answers this through Taylor expansion around a carefully chosen root point $\tilde{\mathbf{x}}$ where $f(\tilde{\mathbf{x}}) = 0$. This root point serves as a neutral reference. In a binary classifier, for instance, it represents a point on the decision boundary where the model assigns equal probability to both classes. The first-order Taylor expansion is:

$$f(\mathbf{x}) \approx \sum_p \frac{\partial f}{\partial x_p}\bigg|_{\tilde{\mathbf{x}}} \cdot (x_p - \tilde{x}_p) \tag{2.9}$$

Each term defines the relevance for input feature p: $R_p = \frac{\partial f}{\partial x_p}|_{\tilde{\mathbf{x}}} \cdot (x_p - \tilde{x}_p)$. This formulation automatically satisfies conservation: $\sum_p R_p = f(\mathbf{x})$.

However, finding a global root point for a deep network is computationally intractable. DTD's crucial innovation is layer-wise decomposition: instead of decomposing the entire function at once, DTD decomposes computation at each layer independently. For each neuron in layer $\ell + 1$ with relevance $R_j^{(\ell+1)}$, DTD applies a local Taylor expansion to

redistribute this relevance to neurons in layer ℓ. The choice of root point determines the resulting propagation rule: choosing the root at the origin yields LRP-0, while choosing roots that zero out only negative contributions yields the z^+-rule.

The DTD framework thus unifies the intuitive conservation principles of LRP with the mathematical rigor of Taylor decomposition. This theoretical foundation proves essential when extending these methods to Transformers, where the interplay of attention mechanisms, skip connections, and normalization layers requires principled handling of relevance flow.

2.3.3 TransLRP: Extending LRP to Transformers

TransLRP [CGW21a] extends Layer-wise Relevance Propagation to Transformer architectures by addressing several methodological challenges: handling skip connections and attention operations without numerical instabilities, supporting non-ReLU activation functions like GELU, and providing class-specific visualizations.

TransLRP propagates relevance backward through each layer operation $L^{(n)}(X, Y)$, which maps inputs X and parameters Y to layer outputs. The generic Deep Taylor propagation rule is:

$$R_j^{(n)} = G(X, Y, R^{(n-1)}) = \sum_i X_j \frac{\partial L_i^{(n)}(X, Y)}{\partial X_j} \frac{R_i^{(n-1)}}{L_i^{(n)}(X, Y)} \tag{2.10}$$

where X and Y are typically the input feature map and weights for layer n, and $L_i^{(n)}(X, Y)$ is the i-th output element of the layer. Gradients are computed with respect to the target class t which can be either the predicted class or any other class for counterfactual analysis. Note that the paper uses reverse indexing where n decreases as we move forward through the network ($n = N$ at input, $n = 1$ at output).

Handling GELU activations. For ReLU networks where activations are non-negative, the LRP rule simplifies to considering only positive values. Since Transformers use GELU activation [HG17] which outputs both positive and negative values, TransLRP modifies the propagation rule by considering only elements with positive weighted relevance: $R_j^{(n)} = \sum_{\{i|(i,j)\in q\}} \frac{x_j w_{ji}}{\sum_{\{j'|(j',i)\in q\}} x_{j'} w_{j'i}} R_i^{(n-1)}$ where $q = \{(i, j)|x_j w_{ji} \geq 0\}$.

Challenges and normalization. Transformer architectures present two unique challenges. Skip connections preserve the conservation rule but can cause numerical instabilities where relevance values can explode even though their sum remains constant. Matrix multiplication in attention does not preserve conservation, leading to double-counting where relevance sums across both inputs equal twice the correct total.

To address these issues, TransLRP introduces a normalization scheme for binary operators that distributes total relevance proportionally between input branches based on their absolute magnitudes, ensuring each branch's relevance sums to the correct total while maintaining conservation and bounding individual relevance sums.

Weighted attention relevance. The key contribution of TransLRP is combining attention scores with both gradients and relevance for class-specific visualization. For each Transformer block b, they compute:

$$\bar{A}^{(b)} = I + E_h(\nabla A^{(b)} \odot R^{(n_b)})^+ \tag{2.11}$$

where $A^{(b)}$ is the attention map, $\nabla A^{(b)}$ are gradients with respect to target class t, $R^{(n_b)}$ are relevance scores at the attention layer (specifically computed at the softmax operation), $\odot$ denotes element-wise multiplication, $(\cdot)^+$ keeps only positive values, E_h averages across attention heads, and I is the identity matrix. The final visualization is obtained by aggregating across all B blocks: $C = \bar{A}^{(1)} \cdot \bar{A}^{(2)} \cdot \ldots \cdot \bar{A}^{(B)}$. The attribution map is extracted from row 0 of the relevance matrix, representing how much each image patch contributed to the classification decision.

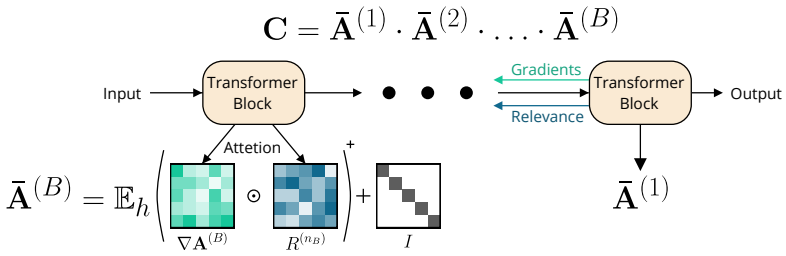


Figure 2.3: TransLRP method overview. Gradients and relevancies are propagated through the network and integrated to produce final relevancy maps. Figure reproduced from [CGW21a].

Computational overhead. While TransLRP demonstrates superior performance, it requires propagating relevance through all layers of the Transformer including attention layers but also MLP blocks, normalization layers, and skip connections. This necessitates custom hooks throughout the entire model to compute relevance scores at each layer, apply normalization at skip connections, and store intermediate values. Despite the computational cost, TransLRP was at the time of release the only method that produced truly class-specific visualizations by design (Figure 2.4).

2.3.4 TransMM: Simplifying Attribution

TransMM (Generic Attention-model Explainability), introduced by Chefer et al. [CGW21b] as an evolution of TransLRP, represents a significant simplification in computing attribution maps for Transformers. While maintaining comparable or superior performance, TransMM eliminates the need for complex LRP propagation entirely.

Key innovation: Eliminating LRP. The fundamental breakthrough of TransMM is demonstrating that expensive relevance propagation through all network layers is unnecessary. While TransLRP requires custom implementation of relevance propagation rules for every layer type, specialized handling of non-ReLU activations, complex normalization

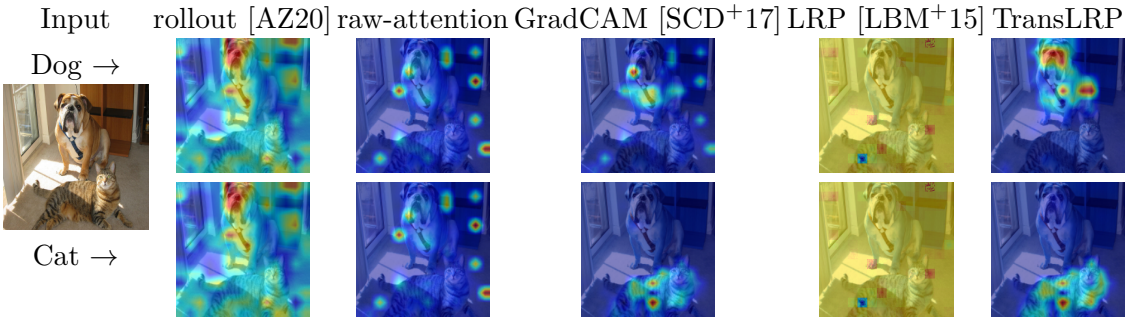


Figure 2.4: Comparison of attribution methods for class-specific visualization. TransLRP produces distinct, well-localized attribution maps for different classes, while rollout, raw-attention, and LRP variants generate identical attributions regardless of target class. Figure reproduced from [CGW21a].

schemes for skip connections, and relevance scores computed through the entire network, TransMM achieves equivalent results using only attention maps and standard gradients.

Mathematical framework. Consider a vision transformer with N image patches and a CLS token for classification. TransMM maintains a relevancy matrix $\mathcal{R}^{(\ell)} \in \mathbb{R}^{(N+1)\times(N+1)}$ at each layer ℓ, where $\mathcal{R}_{ij}^{(\ell)}$ represents the relevance of token j to token i at layer ℓ. The relevancy matrix is initialized as the identity: $\mathcal{R}^{(0)} = I^{(N+1)\times(N+1)}$.

The critical difference between TransLRP and TransMM lies in how attention maps are weighted. TransLRP uses $\bar{A}^{(\ell)} = I + \mathbb{E}_h[(\nabla A^{(\ell)} \odot R^{(n_\ell)})^+]$, requiring LRP relevance scores, while TransMM uses $\bar{A}^{(\ell)} = I + \mathbb{E}_h[(\nabla A^{(\ell)} \odot A^{(\ell)})^+]$, using only raw attention weights. Here $\nabla_{A^{(\ell)}} y_t$ is the gradient of the target logit with respect to attention weights, $\odot$ denotes element-wise multiplication, $(\cdot)_+ = \max(0, \cdot)$ keeps only positive contributions, $\mathbb{E}_h$ averages across attention heads, and I is the identity matrix.

The relevancy matrix is updated through self-attention layers according to:

$$\mathcal{R}^{(\ell)} = \mathcal{R}^{(\ell-1)} + \bar{A}^{(\ell)} \cdot \mathcal{R}^{(\ell-1)} \tag{2.12}$$

The attribution score for each image patch is extracted from the CLS token's row: $\text{Attribution}_i = \mathcal{R}_{0,i}^{(L)}$ for $i \in \{1, \ldots, N\}$.

Practical advantages. The simplification from TransLRP to TransMM has profound practical implications: TransMM needs only attention maps and their gradients with simple hooks at attention layers, requires only a single backward pass for gradients followed by simple matrix operations, and stores only attention maps and the relevancy matrix. Despite this dramatic simplification, Chefer et al. [CGW21b] demonstrate that TransMM achieves comparable or superior performance to TransLRP across multiple benchmarks.

Summary. This section has traced the evolution of attribution methods for Vision Transformers, from the theoretical foundations of LRP and DTD to the practical simplifications of TransMM. The key insights include: traditional attribution methods fail to capture Transformer-specific operations, LRP provides principled relevance propagation with theoretical backing from DTD, TransLRP successfully extends LRP to Transformers but with significant overhead, and TransMM achieves comparable results through dramatic simplification. The remaining challenges include disentangling feature-level contributions within gradient signals and developing theoretical frameworks that better capture Transformer-specific operations. These challenges motivate our investigation of whether TransMM's principle of combining activation patterns with gradient information extends to sparse semantic features.

CHAPTER 3

Interpretability and Faithfulness Metrics

Understanding what neural networks learn and whether attribution methods accurately reflect their reasoning are complementary challenges. This chapter presents the mechanistic interpretability tools that decompose model internals into interpretable features (Section 3.1), then introduces faithfulness metrics that evaluate attribution quality (Section 3.2).

3.1 Mechanistic Interpretability: Understanding Model Internals

Understanding what neural networks learn has been a central challenge since their inception. Early work on CNNs showed promise: individual neurons in trained networks appeared to respond to interpretable visual patterns like edge detectors in early layers [ZF14], texture patterns in middle layers, and object parts in deeper layers [OMS17]. Feature visualization techniques [OMS17, YCN+15] synthesized images that maximally activated specific neurons, often revealing seemingly clear concepts like dog faces, wheels, or text patterns.

However, this interpretability broke down at scale. Most neurons in large networks activated for multiple, seemingly unrelated patterns, a phenomenon termed **polysemanticity** [OCS+20]. Cammarata et al.'s analysis of "curve detectors" [CGC+20] demonstrated this concretely: neurons that appeared to detect curves also responded to high-frequency patterns, 3D object boundaries, and other seemingly unrelated visual features. The promise that individual neurons correspond to interpretable concepts proved insufficient for understanding modern networks.

The word embedding literature provided a suggestive alternative perspective. Mikolov et al.'s word2vec [MCCD13] demonstrated that semantic concepts could exist not in individual dimensions but in *directions* through vector space, observing that "king" - "man" + "woman" ≈ "queen". This raised a possibility: perhaps neural networks encode features in directions through activation space rather than in individual neurons, explaining why neuron-level interpretability fails.

The superposition hypothesis [EHO+22] formalized this intuition: networks exploit activation sparsity to represent more features than they have neurons, storing multiple features in overlapping, non-orthogonal directions. This reframed polysemanticity from a failure of learning into an optimal strategy given capacity constraints. If features exist in superposition rather than in individual neurons, new methods were needed to disentangle them. This poses the motivation behind the development of Sparse Autoencoders.

Beyond SAEs, researchers have developed complementary interpretability tools. Linear probes [AB16] test whether specific information is linearly accessible from layer activations, though they require knowing what concepts to probe for. Activation patching methods [WVC+23, CMPL+23] identify causal circuits by intervening on specific activations and measuring behavioral changes. Each method offers different trade-offs between interpretability, causal understanding, and prior knowledge requirements.

Connection to attribution. Mechanistic interpretability tools complement the attribution methods presented in Chapter 2.3. While attribution answers "which image regions drive predictions," mechanistic interpretability answers "what semantic concepts does the model recognize in those regions." Understanding that a model detected specific SAE features, for instance lung infiltrates, tissue textures, or anatomical boundaries, it provides semantic grounding for attribution maps that would otherwise be abstract heatmaps. This connection enables us to interpret why certain attributions are faithful: they highlight regions where the model activated features causally relevant to classification.

3.1.1 The Superposition Hypothesis

Elhage et al. [EHO+22] provided a theoretical framework for understanding polysemanticity through the **superposition hypothesis**: neural networks exploit the sparsity of feature activation patterns to encode more features than they have dimensions. Through carefully designed toy experiments, they demonstrated not only that superposition emerges naturally under realistic constraints, but also that features in superposition can perform meaningful computation.

Empirical demonstration. Elhage et al. [EHO+22] demonstrated superposition through controlled experiments with single-layer networks trained on synthetic data with varying feature sparsity and importance. They showed that as feature sparsity increases, networks transition from dedicating orthogonal dimensions to the most important features to encoding many features in overlapping directions resulting in a sharp phase transition. Remarkably, features in superposition retain computational capacity, suggesting polysemanticity is not merely a compression artifact but a functional computational

strategy. The transition point depends critically on the relative importance of features: uniformly important features superpose more readily than those with varied importance, explaining why real neural networks trained on naturally varying data ubiquitously exhibit polysemantic neurons.

Key findings on superposition emergence. The experiments revealed a sharp phase transition in network behavior as sparsity increases. When features are dense, the network dedicates orthogonal dimensions to the most important features, leaving others unrepresented. However, as sparsity increases, the network discovers it can exploit the low probability of simultaneous feature activation to encode many features in the same space through superposition. Remarkably, features in superposition retain computational capacity where networks can perform meaningful operations on superposed features, suggesting that polysemanticity is not only a compression artifact but rather a functional computational strategy.

The transition point depends critically on the relative importance of features: uniformly important features superpose more readily than those with varied importance. This finding has profound implications for understanding why real neural networks, trained on data with naturally varying feature importance and sparsity, ubiquitously exhibit polysemantic neurons. The superposition hypothesis fundamentally reframes the challenge of neural network interpretability, revealing superposition as an optimal strategy given limited model capacity and sparse activation patterns.

3.1.2 Sparse Autoencoders: Reversing Superposition

Classical dimensionality reduction methods such as PCA are constrained to extracting at most N orthogonal components from N-dimensional data. However, the superposition hypothesis demonstrates that neural networks encode $M \gg N$ features within N-dimensional activation spaces through exploitation of feature sparsity [EHO+22]. As these features exist in non-orthogonal superposition, recovering interpretable features requires methods capable of learning overcomplete representations.

Sparse Autoencoders (SAEs) [HCS+24, SBB22] address this by learning to reconstruct neural activations through an expanded sparse representation. The method performs an approximate inverse of the superposition process: dense, polysemantic activation vectors are mapped to sparse vectors in a higher-dimensional space where each dimension can correspond to a single semantic concept.

Architecture and training objective. Given an activation vector $\mathbf{x} \in \mathbb{R}^n$ from a neural network layer, an SAE learns an overcomplete dictionary of features through encoding ($\mathbf{f}(\mathbf{x}) = \text{ReLU}(W_{\text{enc}}\mathbf{x} + \mathbf{b}_{\text{enc}})$ where $W_{\text{enc}} \in \mathbb{R}^{m \times n}$ with $m \gg n$) and decoding ($\hat{\mathbf{x}} = W_{\text{dec}}\mathbf{f}(\mathbf{x}) + \mathbf{b}_{\text{dec}}$). The ratio m/n is the expansion factor, typically 8 to 64. This expansion creates enough dimensions to represent each feature with its own orthogonal basis vector, effectively unpacking the compressed representation.

The SAE training objective balances reconstruction and sparsity:

$$\mathcal{L} = \|\mathbf{x} - \hat{\mathbf{x}}\|_2^2 + \alpha\|\mathbf{f}(\mathbf{x})\|_1 \tag{3.1}$$

The reconstruction term ensures the SAE preserves information from original activations, while the sparsity term (L1 norm summing absolute values of all feature activations) encourages most features to remain inactive. The hyperparameter α controls the sparsity-reconstruction tradeoff: higher values produce sparser representations with cleaner feature separation but potentially worse reconstruction, while lower values allow denser representations that may retain some polysemanticity.

From theory to practice. While theoretical foundations were established through controlled experiments, practical value remained uncertain until recent breakthrough applications. Anthropic's work on monosemanticity [BTB+23, TCM+24] demonstrated that SAEs could extract interpretable features from production-scale language models, discovering features ranging from specific tokens to abstract concepts like deception and bias. These features proved as causally relevant where interventions on specific features predictably altered model outputs, validating that SAEs capture genuine computational units rather than statistical artifacts. This success motivated applications to vision transformers, while methodological advances like TopK activation functions [GDlTT+25] addressed common failure modes and provided more stable training dynamics.

Current limitations. Despite their successes, SAEs face significant limitations. Current SAEs fail to capture complete model behavior as shown in one experiment where SAE reconstructions replace model activations in GPT-4 and performance drops to that of a model trained with only 10% of the original compute [GDlTT+25]. Scaling SAEs to large models presents severe computational challenges: each layer requires its own SAE with expansion factor 8-64, making comprehensive SAE analysis of production-scale models economically infeasible. Theoretically, no formal framework proves that neural networks actually implement superposition or that SAE features correspond to the model's true computational units. Most critically, SAEs decompose activations but not the computational mechanisms that produce them. In an analysis by Sharkey et al. [SCB+25] it was argued how these identified directions in activation space are merely the inputs and outputs of neural network computations, not the mechanisms themselves.

These limitations suggest that while SAEs represent progress in making neural networks more interpretable, they may be fundamentally insufficient for complete mechanistic understanding. However, perfect understanding may not be necessary for practical progress. The features SAEs discover, while potentially incomplete, consistently prove to be semantically meaningful and causally relevant, driving adoption across language and vision domains.

3.1.3 SAE Applications in Vision Models

Despite theoretical limitations, SAEs have proven remarkably effective for practical applications in vision models, with recent work demonstrating that SAE features capture

semantically meaningful and causally relevant representations.

SAE features are not only correlational but causally influence model behavior. Templeton et al. [TCM+24] first demonstrated this in language models, while Joseph et al. [Ano25] extended these findings to vision, showing that activating or suppressing specific SAE features in CLIP models predictably alters predictions and can improve robustness. This causal relevance makes SAE features promising candidates for enhancing attribution methods.

Lim et al. [LCCS25] analyzed SAE features in CLIP vision transformers, revealing hierarchical organization from low-level patterns (colors, textures) in early layers to semantic concepts in late layers. Their findings that features are spatially localized and that only top-50 to top-100 features per class maintain classification accuracy suggest SAE features capture causally important semantic concepts, motivating their use in attribution methods.

Stevens et al. [SCBWS25] demonstrated that SAE features can validate attribution methods through targeted intervention. By identifying which features are active in regions highlighted by attribution methods like Grad-CAM, then manipulating those features to observe behavioral changes, they provide experimental evidence for attribution hypotheses. While Stevens et al. use SAEs to validate attributions post-hoc, their work suggests a deeper integration: if SAE features provide semantic understanding that can diagnose attribution quality, could these features be incorporated directly into attribution computation? This question motivates our approach, where we leverage SAE features not just to validate attributions after generation, but to enhance their faithfulness during computation itself.

3.2 Faithfulness: Evaluating Attribution Quality

Attribution methods aim to identify which input regions drive model predictions, but how do we know if these explanations are trustworthy? A visually plausible heatmap may highlight regions that merely correlate with predictions rather than cause them [AGM+18]. In high-stakes domains like medical imaging, where clinicians rely on explanations to verify that models have learned clinically meaningful patterns, faithful attribution is essential for meaningful decision support.

A faithful explanation accurately reflects the model's actual reasoning process and the computational mechanisms that produced the prediction [JG20]. However, we cannot directly observe these mechanisms in deep neural networks. This creates a fundamental evaluation challenge: how do we verify that an attribution correctly represents reasoning we cannot directly observe? The research community has not converged on a single formal definition of faithfulness [JG20]. Different use cases require different properties: correlation with performance changes [BWM20], correct ranking of feature importance [LBM+15], or differences in attribution between regions [WKT+24b] all represent valid but distinct notions. For our purposes, we focus on whether attributions reflect causal

relationships: if an attribution claims a region is important, removing that region should impact the prediction accordingly.

Given that we cannot directly observe model reasoning, faithfulness metrics approximate it through controlled interventions [SBM+16]: if removing a highly-attributed region causes large prediction changes while removing low-attributed regions causes small changes, this provides evidence that attributions reflect actual causal importance. All three metrics we employ implement this perturbation principle differently, testing complementary aspects of whether attributions capture causal relationships. Faithfulness Correlation tests whether attribution magnitudes predict the magnitude of impact when features are perturbed, Pixel Flipping tests whether removing features in order of attributed importance causes appropriately ordered performance degradation, and SaCo tests whether attribution magnitude differences align with actual impact differences through independent perturbation.

3.2.1 Faithfulness Correlation

Faithfulness Correlation [BWM20] quantifies whether attribution scores accurately reflect features' actual impact on model predictions. The metric measures the correlation between the sum of attribution scores for a subset of features and the change in model output when those features are replaced with baseline values.

Given a predictor f, explanation function g, input x, and subset size $|S|$, faithfulness is defined as:

$$\mu_F(f, g; x) = \text{corr}_{S \in \binom{[d]}{|S|}} \left(\sum_{i \in S} g(f, x)_i, f(x) - f(x[x_S = \bar{x}_S]) \right) \tag{3.2}$$

where $x[x_S = \bar{x}_S]$ represents the input with subset S replaced by baseline values $\bar{x}_S$.

Since exhaustively evaluating all $\binom{d}{|S|}$ possible subsets is computationally prohibitive, the metric employs random sampling. For each test point, multiple random subsets of fixed size are selected, their features replaced with baseline values, and the Pearson correlation computed between the sum of attributions and the resulting change in model logits. The choice of baseline $\bar{x}$ significantly impacts results, as shown in the original paper by Bhatt et al. with experiments on zero baseline and average baseline, showing substantial variation between choices. Higher correlation indicates that attribution scores faithfully represent features' actual importance. The metric's reliance on random sampling means it may not capture the full distribution of subset importances, and the correlation measure assumes a linear relationship between attribution sums and output changes, which may not hold for highly non-linear models or when features interact strongly.

3.2.2 Pixel Flipping

Pixel Flipping [LBM+15] provides quantitative validation for attribution techniques by directly testing whether pixels identified as important actually influence model predictions.

Let $\mathbf{x} \in \mathbb{R}^d$ denote an input image and $f : \mathbb{R}^d \to \mathbb{R}^C$ represent the classifier. Given an attribution method A, we compute pixel-wise attribution scores $\{a_i\}_{i=1}^d$.

The procedure follows three steps: compute attribution scores, obtain a permutation π that sorts pixels in descending order of attribution values ($a_{\pi(1)} \geq a_{\pi(2)} \geq \cdots \geq a_{\pi(d)}$), and progressively modify pixels according to this ranking while measuring confidence degradation. For each step k, we construct a perturbed image $\mathbf{x}^{(k)}$ where the top k pixels have been modified, with prediction score $s_k = f(\mathbf{x}^{(k)})_c$ for originally predicted class c.

The faithfulness is quantified by computing the area under the prediction degradation curve:

$$\text{AUC} = \frac{1}{d} \sum_{k=0}^{d-1} \frac{s_k + s_{k+1}}{2} \tag{3.3}$$

A lower AUC indicates faster performance degradation and thus better attribution quality. Bach et al. demonstrate this approach's discriminative power by contrasting the steep degradation when flipping high-attribution pixels against the minimal impact of flipping pixels with near-zero scores.

While Pixel Flipping provides valuable quantitative validation, it suffers from cumulative perturbation limitations [WW22, WKT+24b]. Early pixel removals dominate the metric, and the cumulative nature makes it difficult to isolate individual pixel contributions. When evaluating the least important 10% of pixels, they are only removed after the most important 90% have been eliminated, confounding the assessment of their individual impact. This motivates our use of SaCo alongside Pixel Flipping, as the two metrics capture complementary aspects of faithfulness.

3.2.3 SaCo (Salience-guided Faithfulness Coefficient)

The SaCo metric [WKT+24b] addresses fundamental limitations in existing faithfulness metrics by introducing a principled framework for evaluating whether salience scores accurately reflect pixel contributions to model predictions. The authors identify two core assumptions that underpin faithful explanations: (1) pixels assigned higher salience scores should exert greater influence on predictions compared to those with lower scores, and (2) pixel groups with larger differences in salience scores should exhibit proportionally larger disparities in their actual impacts.

Most existing faithfulness metrics rely on cumulative perturbation where pixels are progressively removed in order of importance. This cumulative strategy inherently conflates the impacts of different pixel groups. When evaluating the least important 10% of pixels, they are only removed after the most important 90% have already been eliminated. Wu et al. demonstrate that this makes it impossible to isolate individual contributions and can lead to misleading evaluations where metrics fail to distinguish between advanced explanation methods and random attribution.

SaCo resolves this by employing individual perturbation. Given an input image with HW pixels ordered by salience scores, the pixels are partitioned into K equally sized

subsets $G_1, G_2, \ldots, G_K$. Each pixel subset is perturbed independently, enabling direct comparison of their influences through:

$$\Delta\text{pred}(x, G_i) = p(\hat{y}(x)|x) - p(\hat{y}(x)|\text{Rp}(x, G_i)) \tag{3.4}$$

where $\text{Rp}(x, G_i)$ denotes the image with pixels in subset G_i replaced by the per-sample mean value.

The metric's key innovation lies in its salience-aware violation testing. For each pair of pixel subsets (G_i, G_j) where $s(G_i) \geq s(G_j)$, SaCo tests whether $\Delta\text{pred}(x, G_i) \geq \Delta\text{pred}(x, G_j)$. Crucially, the evaluation is weighted by the salience difference $|s(G_i) - s(G_j)|$, ensuring that violations are penalized proportionally to the strength of the expectation. The final coefficient is:

$$\text{SaCo} = \frac{\sum_{i<j} w(i,j)}{\sum_{i<j} |w(i,j)|} \tag{3.5}$$

where the weight function incorporates both correctness of ordering and magnitude of salience differences. This formulation produces a coefficient ranging from -1 to 1, where positive values indicate alignment between salience assignments and actual impacts.

Wu et al. demonstrate that SaCo captures a fundamentally different aspect of faithfulness compared to conventional metrics. Among conventionally used metrics (Pixel Flipping, AOPC, LOdds, Comprehensiveness), the average inter-correlation is 0.48, while SaCo correlates only 0.18-0.22 with these metrics, demonstrating it evaluates a complementary aspect of explanation quality. While originally formulated for pixel-level attribution, SaCo naturally extends to patch-based methods like TransMM by redefining pixel subsets as patch subsets. For Vision Transformers operating on N patches, we partition patches into K subsets based on their salience scores. The perturbation operation then replaces entire patches with baseline values, aligning with the natural granularity of transformer-based attribution methods. By addressing the limitations of cumulative perturbation metrics while maintaining complementary evaluation perspectives, SaCo provides comprehensive assessment of whether our feature-gradient approach genuinely improves the alignment between attributed importance and actual model behavior.

CHAPTER 4

Methods: Feature-Gradient Attribution

The fields of *mechanistic interpretability* and *explainable AI* (XAI) share a fundamental goal: understanding how neural networks transform inputs into predictions. Yet, these research traditions have largely evolved in parallel. XAI methods like TransMM achieve strong performance by combining attention patterns with gradient information [CGW21b], but they operate on dense, entangled gradient signals that aggregate influences across all learned representations. Conversely, Sparse Autoencoders (SAEs) successfully decompose polysemantic activations into interpretable features [HCS+24, BTB+23], but have primarily been applied to auditing model internals rather than improving practical explanation methods.

This separation represents a missed opportunity. In this chapter, we bridge this gap by integrating semantic feature information *directly into the attribution computation.*

4.1 Motivation and Core Principle

From Pixels to Semantic Features. While recent work has begun connecting attribution and interpretability [HKK25, SCBWS25], existing approaches lack a mechanism to use semantic structure to calculate attribution itself. Operating in feature space reframes the attribution question: instead of asking "which pixels influence the gradient?", we ask **"which learned semantic concepts drive the decision, and where are they expressed?"**

This provides a principled route to disentangling gradient signals. If raw gradients aggregate influences across all representations, projecting them through a sparse feature basis separates signal from noise by isolating the specific semantic concepts driving each patch's importance.

Intuition: Semantic Decomposition. Consider a patch containing both a dog's eye (relevant) and background texture (irrelevant). A raw gradient at this patch aggregates both signals, failing to distinguish which pattern drives the prediction. However, an SAE trained on the residual stream disentangles these into distinct features, for instance Feature 237 ("eye-like patterns") and Feature 891 ("grass texture").

By projecting the gradient through the SAE, we can calculate a *feature-gradient score*: the product of feature activation (is the concept present?) and feature-space gradient (does the concept influence the output?). This effectively filters the attribution signal: patches where relevant semantic features outweigh irrelevant ones receive high importance, refining the signal at a fundamental level (Figure 4.1).

Extending the TransMM Principle. Our method extends the logic of TransMM to the feature space. TransMM identifies important patches by weighting *attention presence* with *gradient influence*. We apply this same principle to the residual stream: we combine *semantic presence* (SAE feature activations) with *semantic influence* (SAE feature gradients).

We focus on the residual stream as it acts as the primary information pathway in Transformers [ENO+21], and residual stream SAEs have become the standard for reliability and reconstruction quality [BTB+23]. By using this feature-gradient signal to modulate TransMM's attention maps, we create a correction mechanism that preserves TransMM's architectural efficiency while incorporating learned semantic structure.

4.2 Feature-Gradient Decomposition

This section presents the mathematical framework for decomposing residual gradients through SAE feature space and computing per-patch importance scores.

4.2.1 Mathematical Framework

Consider a residual layer ℓ with SAE decoder $D \in \mathbb{R}^{d \times K}$. For each spatial token t:

- Let $x_t \in \mathbb{R}^d$ be the residual vector
- Let $f_t \in \mathbb{R}^K_{\geq 0}$ be the SAE feature activations
- Let $g_t = \nabla_{x_t} y$ be the gradient with respect to the target logit

Feature-space gradient projection. The decoder provides a linear mapping $x_t \approx D f_t$. We project gradients into feature space via:

$$\nabla f_t = D^\top g_t \in \mathbb{R}^K \tag{4.1}$$

where $\nabla f_t^{(k)}$ represents the sensitivity of the target logit to feature k at token t.

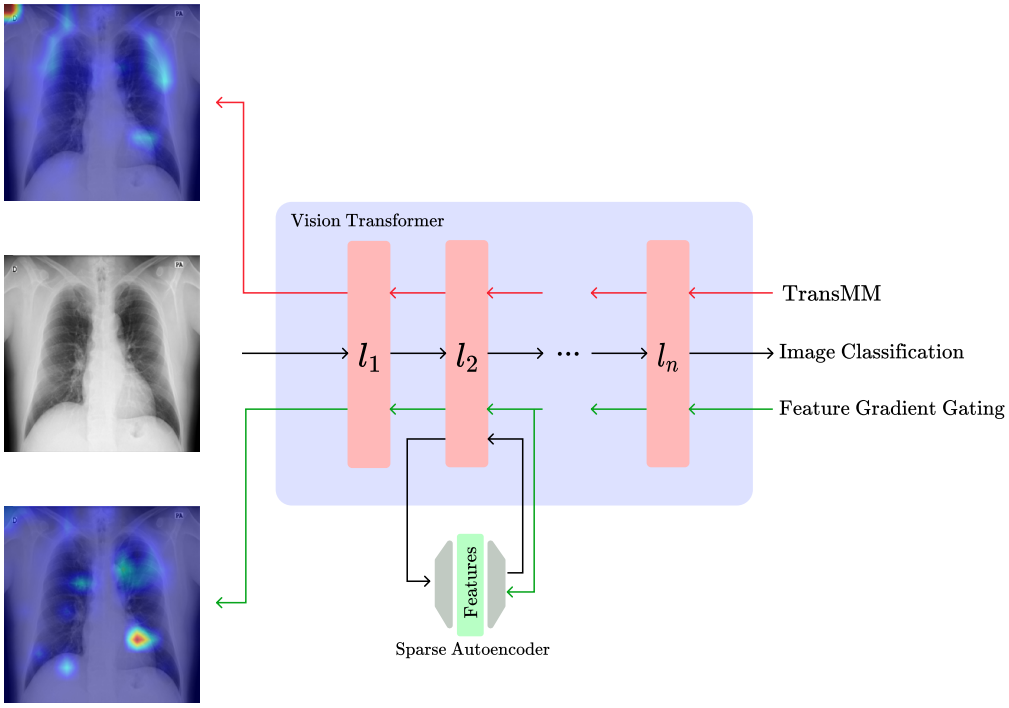


Figure 4.1: Feature-Gradient Attribution overview. Our method extends TransMM by incorporating semantic structure from Sparse Autoencoders (SAEs). Example chest X-rays showing input image (top), vanilla TransMM attribution (middle), and our feature-gated attribution (bottom), demonstrating improved localization of disease-relevant regions.

Feature-weighted scoring. For each feature k, we compute its contribution as the product of gradient sensitivity and activation strength:

$$\bar{f}_t^{(k)} = \nabla f_t^{(k)} \cdot f_t^{(k)} \tag{4.2}$$

We aggregate contributions across all features to obtain a per-patch score:

$$s_t = \sum_{k=1}^{K} \bar{f}_t^{(k)} = \sum_{k=1}^{K} (D^\top g_t)_k \cdot f_t^{(k)} \tag{4.3}$$

This can be interpreted as $\bar{s}_t = g_t^\top (D f_t)$, the dot product between the gradient and the SAE reconstruction.

4.2.2 Method Variants

To understand which aspects of the feature-gradient signal contribute to improved faithfulness, we implement three variant methods. The **combined** method is our primary

contribution, while the activation-only and gradient-only variants serve as ablation studies:

Combined (primary method). Uses both gradient sensitivity and feature activation:

$$s_t^{\text{combined}} = \sum_{k=1}^{K} \nabla f_t^{(k)} \cdot f_t^{(k)} \tag{4.4}$$

Activation-only (ablation). Tests whether feature presence alone provides useful signal:

$$s_t^{\text{activation}} = \sum_{k=1}^{K} f_t^{(k)} \tag{4.5}$$

Gradient-only (ablation). Tests whether gradient projection alone is beneficial:

$$s_t^{\text{gradient}} = \sum_{k=1}^{K} \nabla f_t^{(k)} \tag{4.6}$$

These ablation variants allow us to empirically determine whether the interaction between gradients and activations is necessary, or if either signal alone suffices for improving attribution faithfulness. Figure 4.2 illustrates how these components integrate within a single transformer layer.

4.3 Gate Construction

The per-patch scores s_t must be converted into multiplicative factors that can modulate attention maps. This conversion involves carefully designed steps that respect the sparse activation patterns of SAE features.

4.3.1 Score Normalization

The sparse nature of SAE activations creates a distinctive statistical challenge. In a well-trained sparse autoencoder, the vast majority of features remain inactive (near zero) for any given input, with only a small subset activating strongly [BTB+23]. This sparsity pattern means that for most patches, feature-gradient scores will cluster near zero, representing patches where no particularly relevant semantic features are active.

We normalize scores across spatial tokens using robust statistics. Rather than standard z-score normalization, we use median-based normalization:

$$\hat{s}_t = \frac{s_t - \text{median}(s)}{\text{MAD}(s) + \epsilon} \tag{4.7}$$

where $\text{MAD}(s) = \text{median}(|s_t - \text{median}(s)|)$ is the median absolute deviation, and $\epsilon = 10^{-8}$ prevents division by zero. The MAD is scaled by 1.4826 to approximate standard deviation for normally distributed data while maintaining robustness to outliers.

$$\text{TransMM: } \bar{A}^{(\ell)} = I + \mathbb{E}_h\left[\left(A^{(\ell)} \odot \nabla A^{(\ell)}\right)^+\right]$$

$$\text{Ours: } \bar{A}^{(\ell)}_{\text{gated}} = \bar{A}^{(\ell)} \cdot \text{diag}\left(\text{Gate}(f^{(\ell)} \odot \nabla f^{(\ell)})\right)$$

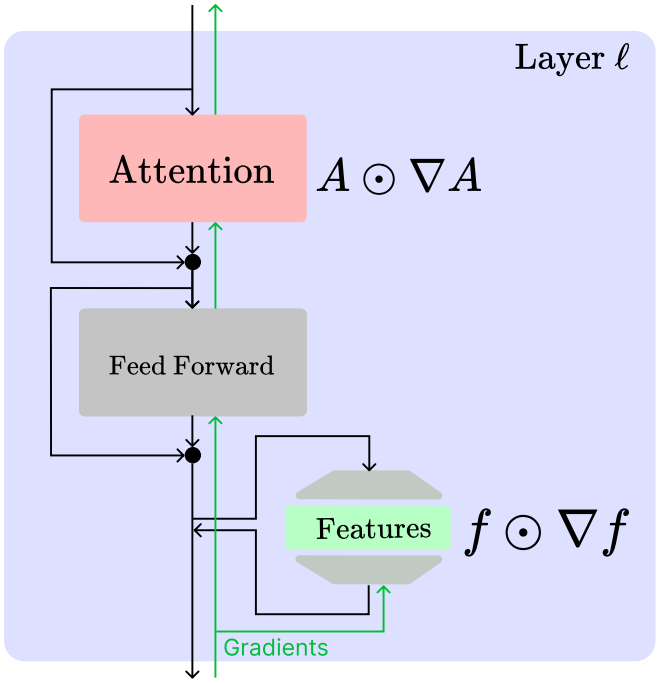


Figure 4.2: Detailed layer-wise computation. A single transformer layer showing where feature-gradient gating occurs. TransMM computes gradient-weighted attention from attention weights (pink). In parallel, we extract residual stream activations and pass them through a trained Sparse Autoencoder, which decomposes activations into interpretable features via an encoder-decoder architecture (green). We project gradients through the SAE decoder to obtain feature-space gradients. The element-wise product captures which semantic features are both present and influential. The Gate function (Section 4.3) aggregates these scores, normalizes using robust statistics, and maps to multiplicative gates that modulate attention before relevancy propagation.

This median-based approach is particularly well-suited to the sparse activation regime. The median naturally identifies the baseline of "not activated" patches where features are weakly or not at all engaged. When features do activate strongly at a patch, the score is correctly normalized relative to this inactive baseline. In contrast, mean-based z-score normalization would be problematic: the mean would be pulled toward the tail of highly active features, causing the baseline to shift and potentially assigning non-neutral gates to patches that should be treated as neutral.

This choice ensures that the normalization reflects the semantic structure of feature activation patterns: patches with typical (median-level) feature-gradient scores receive neutral gates, while only patches with genuinely unusual feature activation patterns receive substantial corrections.

4.3.2 Exponential Mapping to Multiplicative Gates

We map normalized scores to multiplicative gates using an exponential transformation that naturally produces symmetric amplification and suppression centered at 1:

$$w_t = \exp(\log(c_{\max}) \times \tanh(\kappa \cdot \hat{s}_t)) \tag{4.8}$$

where $c_{\max}$ is the maximum gate value and κ is the temperature parameter controlling sensitivity.

This formulation creates a symmetric multiplicative range centered at the multiplicative identity: $w_{\min} = 1/c_{\max}$, $w_{\max} = c_{\max}$, and $w_{\text{center}} = 1.0$ (neutral, no modification). The behavior at different normalized score values is:

- When $\hat{s}_t = 0$ (median score): $\tanh(0) = 0$, so $w_t = 1.0$ (neutral gate)
- When $\hat{s}_t \to +\infty$: $\tanh(\cdot) \to 1$, so $w_t \to c_{\max}$ (maximum amplification)
- When $\hat{s}_t \to -\infty$: $\tanh(\cdot) \to -1$, so $w_t \to 1/c_{\max}$ (maximum suppression)

For example, with $c_{\max} = 10.0$, the gate range becomes [0.1, 10.0], allowing 10× amplification and 0.1× suppression. This exponential formulation provides several key advantages: true multiplicative centering, symmetry in log-space, natural bounding without explicit clipping, controllable sensitivity via κ and graceful saturation for extreme scores.

4.4 Integration with TransMM

This section describes how feature-gradient gates are integrated into TransMM's relevancy propagation framework.

4.4.1 Attention Map Modulation

Recall that TransMM computes gradient-weighted attention as $\bar{A}^{(\ell)} = I + \mathbb{E}_h[(\nabla A^{(\ell)} \odot A^{(\ell)})_+]$. Our method modulates this weighted attention map by applying per-patch gates via diagonal matrix multiplication:

$$\bar{A}^{(\ell)}_{\text{gated}} = \bar{A}^{(\ell)} \cdot \text{diag}(1, w_0, w_1, \ldots, w_{N-1}) \tag{4.9}$$

where the diagonal matrix has the CLS token gate fixed at 1 (no modification) and spatial token gates $\{w_t\}$ computed via feature-gradient decomposition. Right-multiplication by a diagonal matrix scales each column independently, preserving the CLS token's attention distribution while modulating spatial tokens according to their semantic importance.

4.4.2 Modified Relevancy Propagation

The standard TransMM update rule is then applied using the gated attention map:

$$\mathcal{R}^{(\ell)} = \mathcal{R}^{(\ell-1)} + \bar{A}^{(\ell)}_{\text{gated}} \cdot \mathcal{R}^{(\ell-1)} \tag{4.10}$$

The feature-gradient gating can be applied at selected layers to compound the correction effect. When applying gating at a subset of layers $\mathcal{L} \subseteq \{1, \ldots, L\}$:

$$\mathcal{R}^{(\ell)} = \begin{cases} \mathcal{R}^{(\ell-1)} + \bar{A}^{(\ell)}_{\text{gated}} \cdot \mathcal{R}^{(\ell-1)} & \text{if } \ell \in \mathcal{L} \\ \mathcal{R}^{(\ell-1)} + \bar{A}^{(\ell)} \cdot \mathcal{R}^{(\ell-1)} & \text{otherwise} \end{cases} \tag{4.11}$$

Our validation experiments investigate which layers benefit most from this gating mechanism, testing both single-layer and multi-layer configurations.

4.5 Algorithm

Algorithm 4.1 presents the complete procedure for computing feature-gradient enhanced attributions using the combined method.

4.6 Experimental Design

This section describes the datasets, models, SAE training procedures, faithfulness evaluation protocol, and validation strategy used to assess our method empirically. All results are presented in Chapter 5.

4.6.1 Datasets and Models

We evaluate our method on three datasets representing different visual reasoning challenges: two medical imaging tasks requiring fine-grained discrimination of subtle anatomical and pathological features, and one natural image dataset designed to test robustness against spurious correlations.

COVID-QU-Ex: Chest X-ray classification. COVID-QU-Ex [TCK+21] contains 33,920 frontal chest radiographs across three classes: COVID-19 infections (11,956 images), non-COVID pneumonia (11,263 images), and healthy controls (10,701 images). We use the ViT-Base/16 model fine-tuned by [KBB23], which achieved 95.4% test accuracy. We adopt the standard 65/15/20 train/validation/test split.

Hyperkvasir: Gastrointestinal endoscopy. Hyperkvasir [BTS+20] provides multi-class gastrointestinal endoscopy images. We utilize the anatomical landmarks subset focusing on six structures: cecum, ileum, retroflex-rectum (lower GI), and pylorus, retroflex-stomach, z-line (upper GI). We use the ViT-Base/16 model fine-tuned by [SM24] with 80/10/10 train/validation/test split, achieving macro F1-score of 0.828.

Algorithm 4.1: Feature-Gradient Gating for TransMM (Combined Method)

Input: Attention maps $\{A^{(\ell)}\}$, SAEs for layers $\ell \in \mathcal{L}$, input image x, target class t

Output: Enhanced attribution map

Parameters: κ: gate strength, $c_{\max}$: maximum gate value

Initialize $\mathcal{R}^{(0)} = I$;

Compute target logit gradient via backpropagation on y_t;

for *layer* $\ell = 1$ **to** L **do**

  Compute $\bar{A}^{(\ell)} = I + \mathbb{E}_h[(\nabla A^{(\ell)} \odot A^{(\ell)})_+]$; `// Standard TransMM`

  if $\ell \in \mathcal{L}$ **then**

    `// Apply feature-gradient gating`

    Extract residual activations $\{x_t\}$ and gradients $\{g_t = \nabla_{x_t} y_t\}$;

    foreach *spatial token* t **do**

      $f_t \leftarrow \mathrm{SAE}^{(\ell)}.\mathrm{encode}(x_t)$; `// Get sparse features`

      $\nabla f_t \leftarrow (D^{(\ell)})^\top g_t$; `// Project gradients to feature space`

      `// Compute weighted features`

      for $k = 1$ **to** K **do**

        $\bar{f}_t^{(k)} \leftarrow \nabla f_t^{(k)} \cdot f_t^{(k)}$;

      end

      $s_t \leftarrow \sum_{k=1}^{K} \bar{f}_t^{(k)}$; `// Aggregate to per-patch score`

    end

    `// Normalize scores using robust statistics`

    $s_{\text{med}} \leftarrow \mathrm{median}(\{s_t\}_{t=0}^{N-1})$;

    $s_{\text{MAD}} \leftarrow 1.4826 \cdot \mathrm{median}(\{|s_t - s_{\text{med}}|\}_{t=0}^{N-1})$;

    foreach *spatial token* t **do**

      $\hat{s}_t \leftarrow \frac{s_t - s_{\text{med}}}{s_{\text{MAD}} + \epsilon}$; `// Normalize`

      $w_t \leftarrow \exp(\log(c_{\max}) \cdot \tanh(\kappa \cdot \hat{s}_t))$; `// Map to gate`

    end

    `// Apply gates to attention map via diagonal matrix`

    $\bar{A}^{(\ell)}_{\text{gated}} \leftarrow \bar{A}^{(\ell)} \cdot \mathrm{diag}(1, w_0, w_1, \ldots, w_{N-1})$;

    $\bar{A}^{(\ell)} \leftarrow \bar{A}^{(\ell)}_{\text{gated}}$; `// Update for propagation`

  end

  $\mathcal{R}^{(\ell)} \leftarrow \mathcal{R}^{(\ell-1)} + \bar{A}^{(\ell)} \cdot \mathcal{R}^{(\ell-1)}$; `// TransMM relevancy propagation`

end

return $\mathcal{R}^{(L)}[0, 1:N]$; `// CLS token attribution to patches`

ImageNet: Natural image classification. ImageNet [RDS+15] is a large-scale dataset with 1,000 object categories. We use the CLIP ViT-B/32 model trained on DataComp-1B [GIF+23], which was contrastively pre-trained on 1.4 billion image-text pairs and achieves 72.7% zero-shot top-1 accuracy. Due to computational constraints, we evaluate on a randomly selected 10,000-image subset of the test set (fixed seed 42 for reproducibility). This dataset tests whether our method generalizes to diverse natural images with a pre-trained vision-language model.

4.6.2 Sparse Autoencoder Training

A critical component of our method is the availability of high-quality SAEs trained on each model's residual stream activations. We adopt different strategies based on dataset characteristics and available resources.

Training SAEs for medical datasets. For COVID-QU-Ex and Hyperkvasir, we train dataset-specific SAEs to ensure features capture domain-relevant patterns. We employ the Prisma framework [JSH+25], configuring SAEs with 64× expansion factor (mapping 768-dimensional residuals to 49,152 SAE features), Top-K activation (K=128), and standard sparse autoencoder architecture with loss $\mathcal{L} = \|\mathbf{x} - \hat{\mathbf{x}}\|_2^2 + \lambda\|\mathbf{f}\|_1$. We train SAEs for layers 2-10 (intermediate to late representations).

SAEs are trained exclusively on patch token activations, not on the CLS token. Since our attribution method applies gates to spatial patches (Equation 4.9), the CLS token provides no useful training signal. Training on patch tokens ensures learned features reflect semantic content of local image regions rather than the global aggregated representation.

We conduct systematic sweeps over learning rates $\{5\times10^{-4}, 1\times10^{-3}\}$ for 3 epochs with cosine annealing, 500-step linear warmup, batch size 4,096, and Adam optimization [KB15]. For each layer, we select the SAE achieving >95% explained variance while minimizing dead features (<10%). Table 4.1 presents selected configurations.

Utilizing pre-trained SAEs for ImageNet. For ImageNet experiments, we leverage publicly available SAEs trained on CLIP-ViT-B/32 by Prisma [JSH+25]. We use their vanilla spatial-patch SAEs, which demonstrated superior reconstruction quality: 99% explained variance across layers 0-9, consistently low dead feature percentages (<0.1%), and high cosine similarity (0.99) between original and reconstructed activations. These SAEs maintain architectural consistency with our training protocol (64× expansion factor, ReLU activation).

Training SAEs on ImageNet's 1.2M images would require substantial computational resources beyond our experimental scope. The pre-trained Prisma SAEs provide well-validated, high-quality features that enable us to test whether our method generalizes beyond domain-specific medical imaging to diverse natural image classification.

Table 4.1: Selected SAE configurations for medical datasets. All SAEs use expansion factor 64× and Top-K (K=128) activation. Metrics shown are explained variance (%) and dead features (%).

Dataset	Layer	Learning Rate	Explained Var. (%)	Dead Features (%)
COVID-QU-Ex	1	1e-3	96.3	0.0
	2	1e-3	93.8	0.0
	3	1e-3	92.1	0.0
	4	1e-3	90.9	0.0
	5	1e-3	92.1	0.0
	6	1e-3	93.0	0.0
	7	1e-3	93.1	0.0
	8	1e-3	91.9	0.0
	9	1e-3	90.2	0.0
	10	1e-3	89.2	0.0
Hyperkvasir	1	5e-4	98.6	0.0
	2	5e-4	97.4	0.0
	3	5e-4	96.5	0.0
	4	5e-4	95.7	0.0
	5	5e-4	95.2	0.0
	6	5e-4	95.3	0.0
	7	5e-4	95.1	0.0
	8	5e-4	95.3	0.0
	9	5e-4	94.4	0.0
	10	5e-4	92.9	0.0

4.6.3 Faithfulness Evaluation Protocol

To capture faithfulness from multiple dimensions, we employ three complementary perturbation-based metrics that measure different aspects of attribution quality: Faithfulness Correlation measures magnitude alignment between attribution scores and actual impact, Pixel Flipping tests whether removing features in attribution order causes appropriate performance degradation, and SaCo evaluates whether attribution magnitude differences correspond to impact differences through independent perturbation.

Patch-level evaluation. TransMM produces attribution scores at the patch level, one value per image patch, reflecting Vision Transformers' native tokenization structure. We adapt all metrics to operate at patch granularity to respect this architectural reality:

- **Faithfulness Correlation** [BWM20]: We randomly sample subsets of 20 patches, replace them with per-sample mean values, and compute Pearson correlation between attribution sums and logit changes.
- **Pixel Flipping** [LBM$^+$15]: We remove entire patches sequentially in order of attribution importance, replacing each with per-sample mean values, and compute area under the performance degradation curve.

- **SaCo** [WKT+24b]: We partition patches into equally sized groups based on attribution rankings and perturb each group independently by replacing patches with per-sample mean values, computing salience-weighted faithfulness coefficient.

This patch-level evaluation provides a principled assessment of whether attribution methods correctly identify which patches contribute to predictions.

4.6.4 Validation Strategy

We conduct validation experiments to assess feasibility and optimal configurations across three datasets. Our validation strategy consists of three complementary phases:

Single-layer method comparison. Evaluates three method variants (combined, activation-only, gradient-only) at each layer (2-10) to determine whether attribution improvement stems from feature activations, decomposed gradients, or their interaction. For all single-layer experiments, we fix $\kappa = 0.5$ and $c_{\max} = 10.0$ based on preliminary exploration. This phase identifies optimal layers for each dataset and establishes which signal components are necessary for faithful attribution.

Multi-layer configurations. Having identified optimal single layers, we investigate whether combining features from adjacent layers provides synergistic benefits. For each dataset, we test 3-4 layer combinations centered on best-performing single layers using the combined method with $\kappa = 0.5$ and $c_{\max} = 10.0$. This phase determines whether multi-layer feature integration enhances attribution quality beyond single-layer approaches.

Hyperparameter validation. We conduct systematic sweeps over gate strength $\kappa \in \{0.1, 0.5, 1.0\}$ and maximum gate value $c_{\max} \in \{2.0, 10.0, 50.0\}$ on the best-performing multi-layer configuration per dataset (9 combinations per dataset). This phase identifies dataset-specific optimal hyperparameters for test set evaluation, assessing robustness across parameter space.

For all validation experiments, we use random subsets of 500 images from validation sets (full validation set for Hyperkvasir due to smaller size), evaluating faithfulness using SaCo, Faithfulness Correlation, and Pixel Flipping across layers 2-10.

4.6.5 Shuffled Decoder Control

To verify that improvements stem from semantic feature structure rather than statistical artifacts of SAE decomposition, we implement a shuffled decoder control. We randomly permute decoder columns: $D_{\text{shuffled}} = D[:, \pi]$ where π is a random permutation. This control preserves feature activation statistics, reconstruction quality, and gating computations while eliminating the correspondence between features and semantic concepts.

If semantic structure is essential, real features should substantially outperform shuffled decoders. If only statistical properties matter (e.g., sparsity-induced denoising), performance should be comparable. This control provides critical validation that observed

improvements require semantic alignment between features and visual concepts, not merely sparse decomposition properties.

4.6.6 Statistical Testing

To assess whether observed performance differences are statistically significant, we employ Welch's t-test, which compares means without assuming equal variances between methods. We compute two-tailed p-values comparing each method's test set performance against the vanilla TransMM baseline. Significance levels are denoted as: $^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$.

4.7 Summary

We have presented Feature-Gradient Attribution, a method that enhances TransMM by incorporating semantic information from learned sparse features. Our approach:

- **Provides a principled integration**: Projects residual gradients through SAE feature space and converts to multiplicative gates via carefully designed normalization and exponential mapping that respects sparse activation patterns
- **Maintains practical advantages**: Operates as a lightweight modulation of TransMM's attention maps, preserving architectural simplicity
- **Enables systematic investigation**: Multiple ablation variants, shuffled decoder controls, and comprehensive evaluation protocols allow us to isolate the contributions of different signal components

The method's effectiveness, including optimal hyperparameters, layer selections, and validation of semantic structure necessity is determined through comprehensive empirical evaluation presented in Chapter 5.

CHAPTER 5

Experimental Results

This chapter presents experimental validation of Feature-Gradient Attribution across three phases. Section 5.1 compares method ablations (combined, activation-only, gradient-only) across individual layers to determine which signal components drive attribution improvement. Section 5.2 investigates whether combining features from adjacent layers provides synergistic benefits. Section 5.3 systematically explores gate strength and clipping range to identify optimal configurations. Finally, Section 5.4 evaluates the best validation configurations on held-out test sets with shuffled decoder controls. All experimental design details, including datasets, SAE training, and faithfulness metrics, are described in Chapter 4.

5.1 Single-Layer Method Comparison

We evaluate three method variants at each layer (2-10) to determine whether attribution improvement stems from feature activations, decomposed gradients, or their interaction. Figures 5.1–5.2, 5.4–5.3, and 5.5–5.6 present performance across all faithfulness metrics.

5.1.1 Key Findings

Combined method consistently superior. The combined method (blue lines) outperforms vanilla TransMM baseline (red dashed) across all datasets, layers, and metrics. Peak improvements occur at intermediate layers:

- **COVID-QU-Ex:** Layers 3-5 show 4-16% SaCo, 10-17% faithfulness correlation, and 1-4% pixel flipping improvements
- **Hyperkvasir:** Layers 4-7 achieve 16-31% SaCo, 7-14% faithfulness correlation, and 2-3% pixel flipping gains

- **ImageNet:** Layers 5-10 demonstrate 16-40% SaCo, 3-10% faithfulness correlation, and 2-5% pixel flipping enhancements

Gradient-only systematically underperforms. Critically, the gradient-only variant (green lines) performs at or below vanilla baseline across most layers and metrics. On the SaCo metric the gradient only method performs over baseline and better than the activation only method. For the Hyperkvasir dataset it even outperforms the combined method. Generally, for all other metrics at all other datasets there is no visible improvement over activation only.

Activation-only shows moderate success. The activation-only variant (orange lines) generally matches or slightly exceeds vanilla baseline, with modest improvements (2-6% SaCo on most layers). This suggests feature activations alone carry some attribution signal, but the combined method's substantial superiority demonstrates that gradient information—when properly grounded through semantic features—significantly enhances attribution quality.

Layer-dependent performance. Optimal layers vary by architecture and training objective. Medical datasets (fine-tuned ViT-B/16) peak at layers 4-6, while ImageNet (CLIP ViT-B/32) peaks at layers 5-10 with broader optimal range. Early layers (2-3) generally underperform, likely encoding low-level textures insufficient for classification-relevant attribution. Late layers (8-10) show variable performance, potentially due to increased feature non-locality as observed in SAE interpretability research [HKK25].

Metric complementarity. Different metrics occasionally peak at different layers (e.g., Hyperkvasir combined method: SaCo peaks at layer 4 while some faithfulness correlation peaks occur at layer 6). This suggests that our metrics capture distinct faithfulness aspects, and robust configurations should perform well across multiple metrics.

Figure 5.1: COVID-QU-Ex single-layer performance: Faithfulness Correlation and SaCo. The combined method (blue) consistently outperforms vanilla baseline (red dashed), peaking at layers 3-5. Gradient-only (green) shows systematic underperformance in late layers.

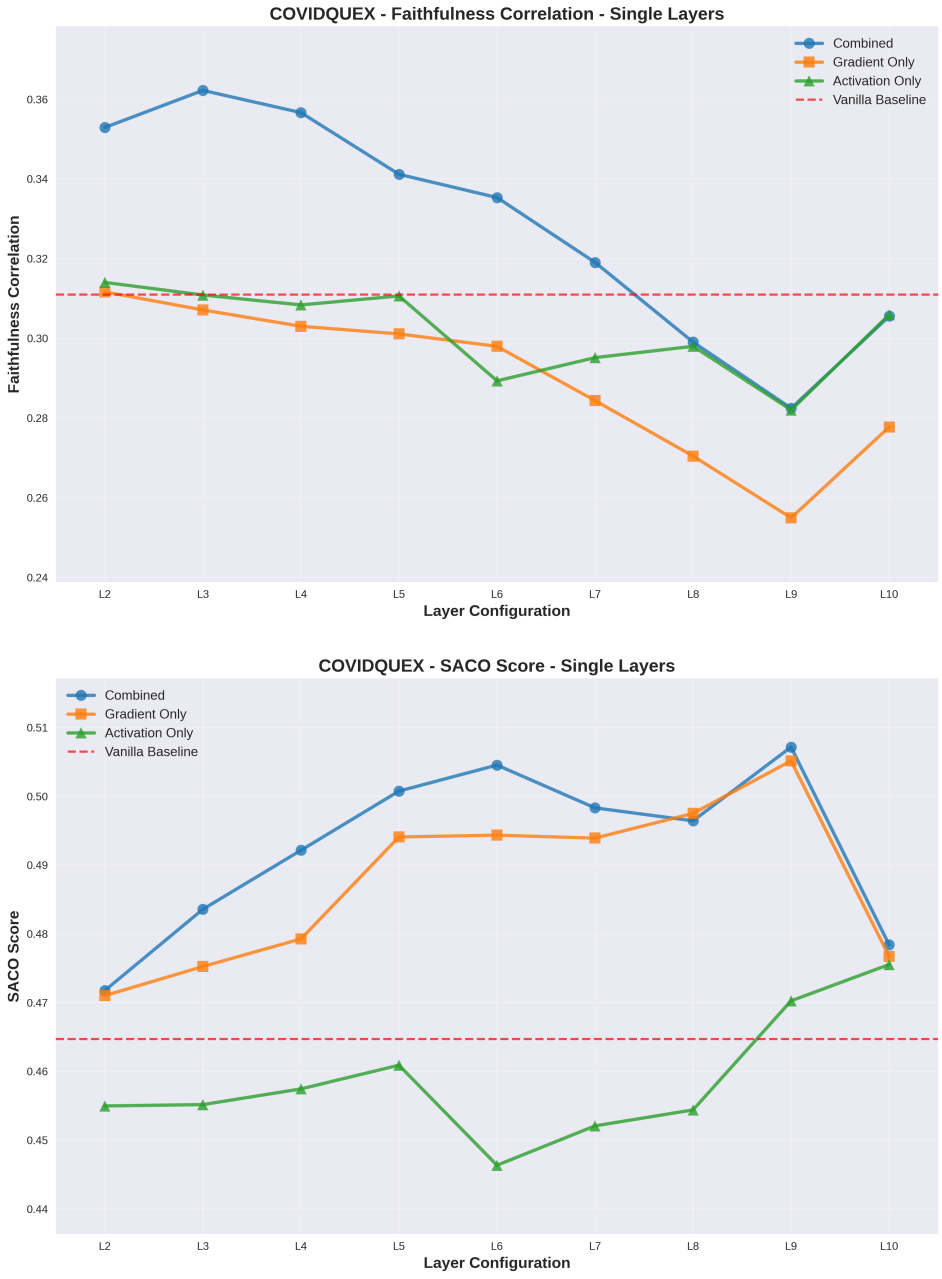


Figure 5.2: COVID-QU-Ex single-layer performance: Pixel Flipping. The combined method maintains consistent improvements, with activation-only (orange) showing moderate performance. Results support the gradient entanglement hypothesis.

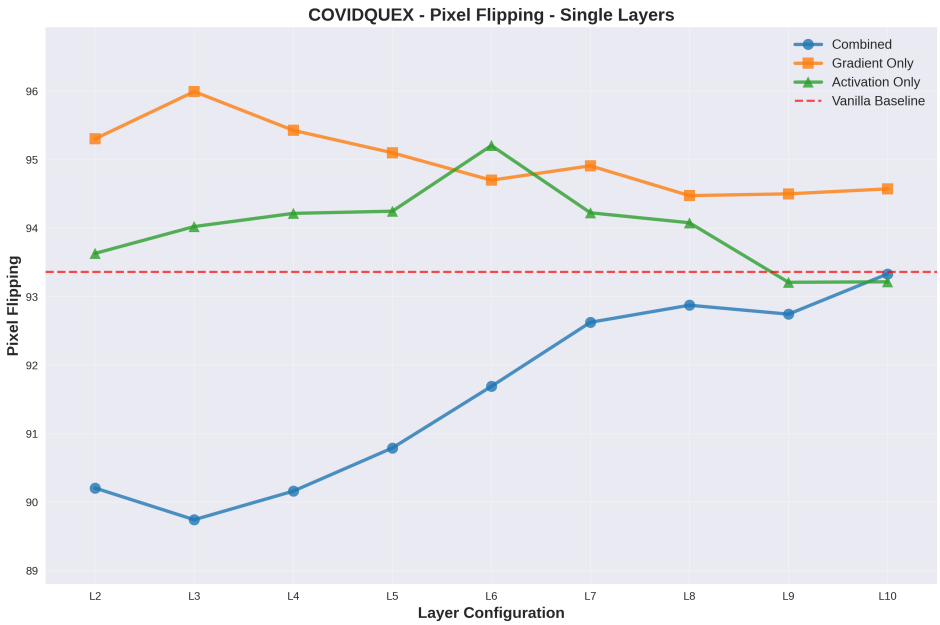


Figure 5.3: Hyperkvasir single-layer performance: Pixel Flipping. The combined method demonstrates consistent improvements across the optimal layer range, confirming the benefits of feature-gradient integration.

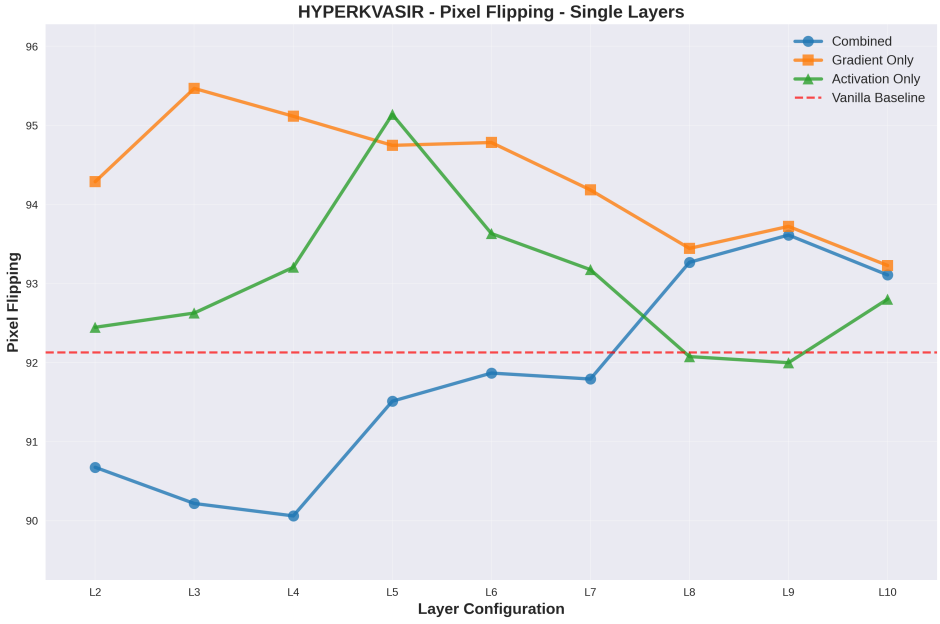


Figure 5.4: Hyperkvasir single-layer performance: Faithfulness Correlation and SaCo. Optimal layers 4-7 show strong combined method improvements. Activation-only (orange) shows moderate performance, while gradient-only degrades faithfulness in late layers.

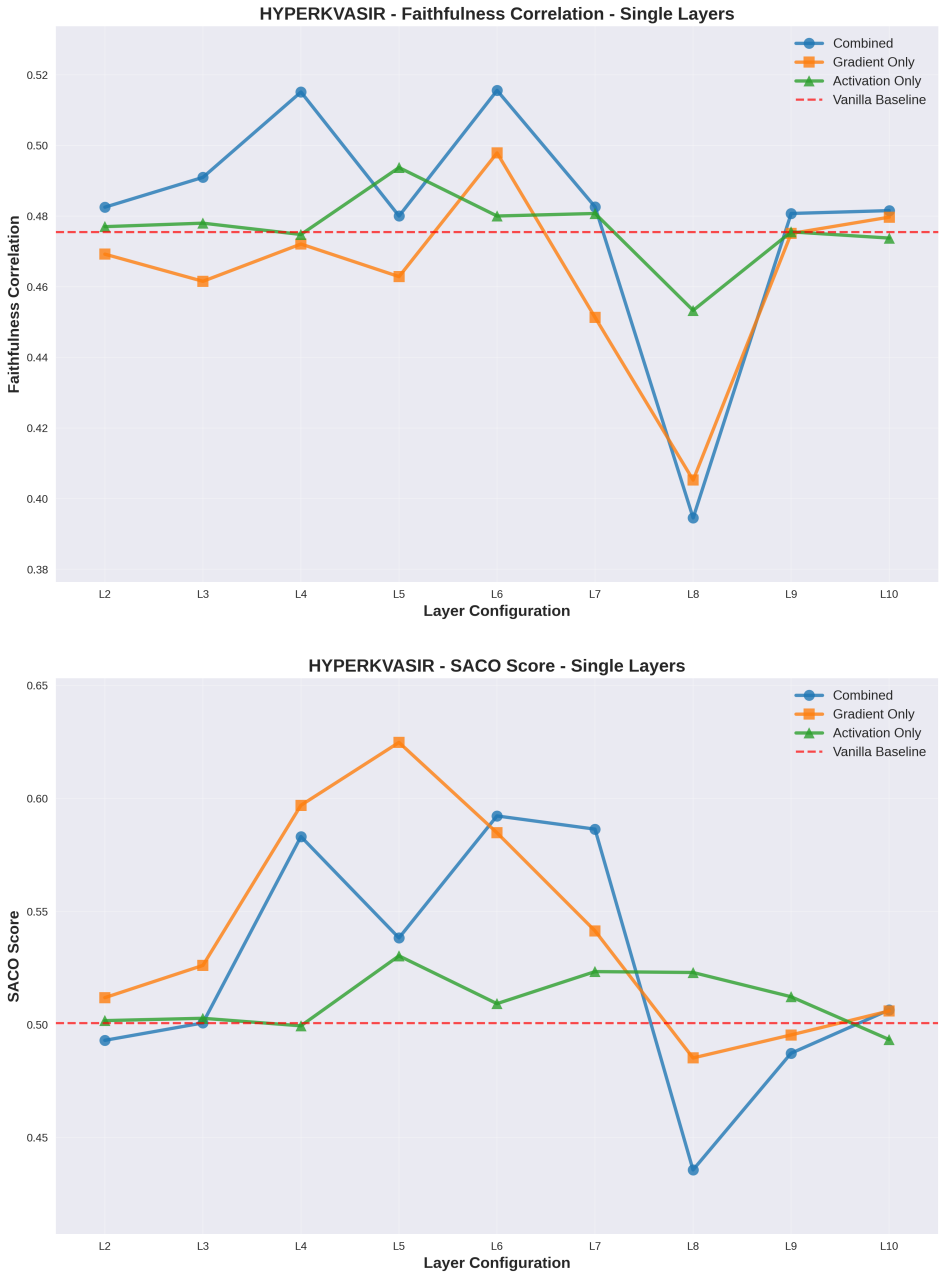


Figure 5.5: ImageNet single-layer performance: Faithfulness Correlation and SaCo. CLIP-based model shows broader optimal layer range (5-10) compared to medical datasets. Combined method maintains consistent improvements despite architectural differences (ViT-B/32 vs. B/16).

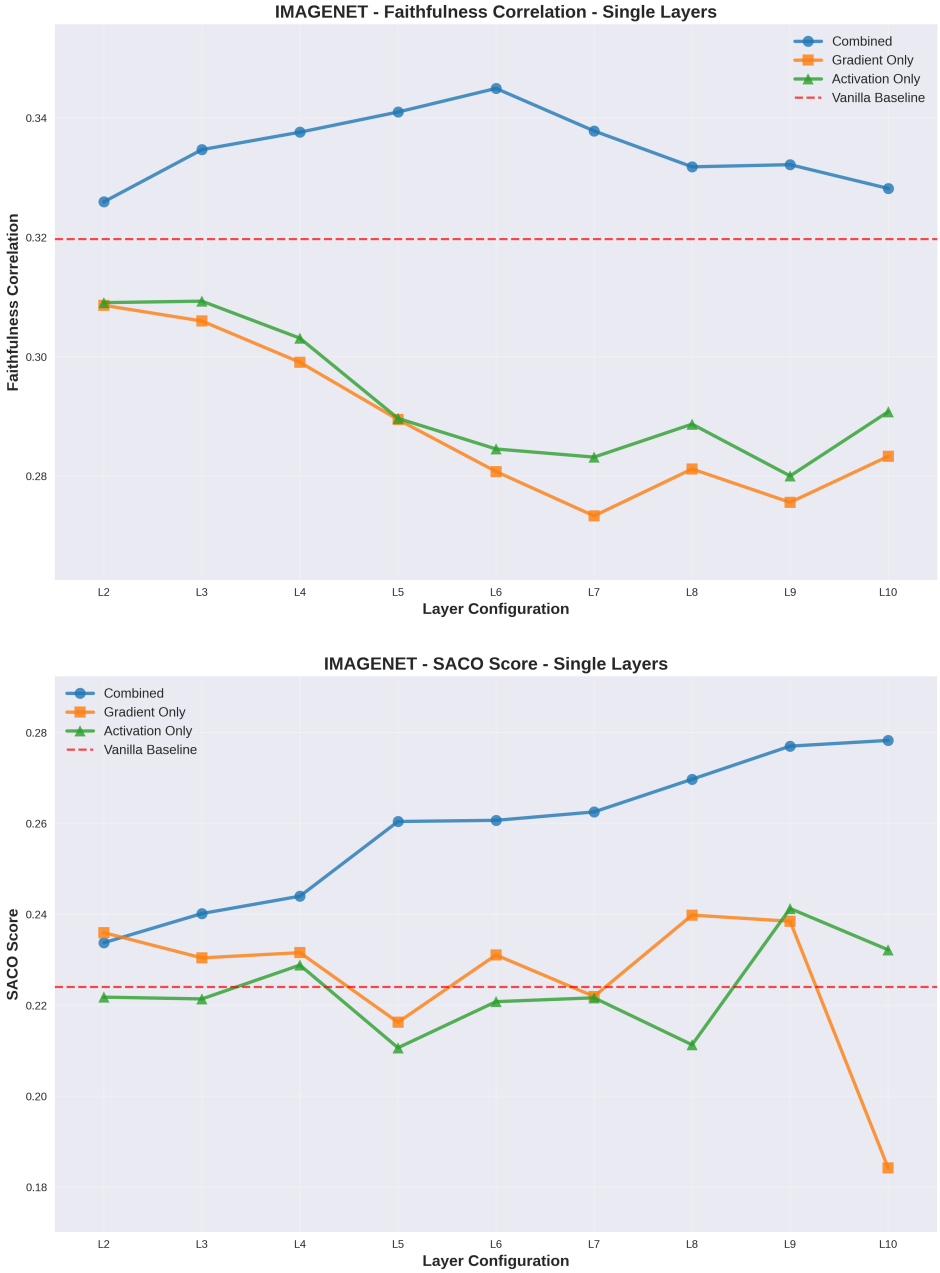
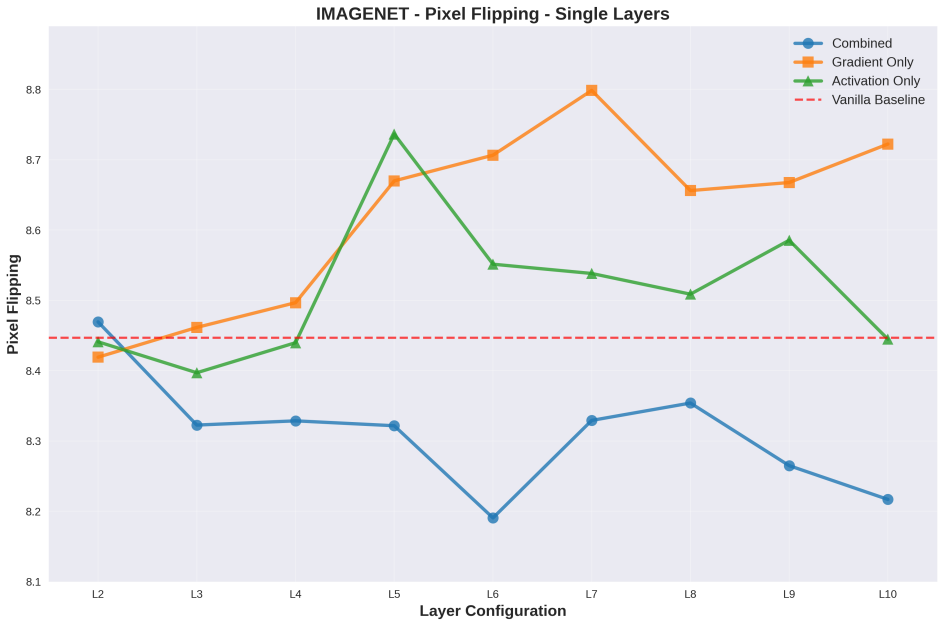


Figure 5.6: ImageNet single-layer performance: Pixel Flipping. The combined method shows consistent improvements, with the broader optimal layer range reflecting differences in contrastive pre-training versus supervised fine-tuning.



5.2 Multi-Layer Configurations

Having identified optimal single layers, we investigate whether combining features from adjacent layers provides complementary semantic information. We evaluate multi-layer configurations using the combined method with $\kappa = 0.5$ and $c_{\max} = 10.0$.

For each dataset, we test 3-4 layer combinations centered on the best-performing single layers:

- **COVID-QU-Ex:** [2,3,4], [3,4,5], [3,6,9], [4,5,6]
- **Hyperkvasir:** [4,5,6], [4,6,7], [5,6,7]
- **ImageNet:** [4,6,8], [5,6,9], [5,6,10], [5,6,7,9], [6,9,10]

Figures 5.7–5.9 compare multi-layer results against single-layer baselines.

5.2.1 Key Findings

Multi-layer synergy confirmed. Multi-layer configurations consistently outperform the best single-layer results across all datasets:

- **COVID-QU-Ex [2,3,4]:** 10.03% SaCo improvement vs. 4.06% best single-layer (layer 3); 32.30% faithfulness correlation improvement vs. 16.48% single-layer; 7.85% pixel flipping improvement vs. 3.88% single-layer
- **Hyperkvasir [4,6,7]:** 30.92% SaCo improvement vs. 16.48% best single-layer (layer 4); 13.99% faithfulness correlation improvement vs. 8.35% single-layer; 2.55% pixel flipping improvement vs. 2.25% single-layer
- **ImageNet [5,6,10]:** 39.96% SaCo improvement vs. 23.63% best single-layer (layer 9); 11.69% faithfulness correlation improvement vs. 3.89% single-layer; 3.94% pixel flipping improvement vs. 2.15% single-layer

This synergy suggests that features from adjacent layers capture complementary semantic aspects, where earlier layers may encode foundational patterns that contextualize higher-level concepts in later layers. The multi-layer improvements are particularly dramatic for COVID-QU-Ex faithfulness correlation (nearly 2× the single-layer gain) and ImageNet SaCo (1.8× the single-layer gain).

Layer selection matters. Not all multi-layer combinations provide equal benefit. The best-performing multi-layer configurations use adjacent or near-adjacent layers: [2,3,4] for COVID-QU-Ex, [4,6,7] for Hyperkvasir, and [5,6,10] for ImageNet. Non-adjacent layer combinations (e.g., COVID-QU-Ex [3,6,9]) generally underperform these optimal adjacent combinations, suggesting semantic continuity across nearby layers is important for effective feature integration.

Figure 5.7: COVID-QU-Ex multi-layer configuration performance. Bar charts compare single-layer peaks against multi-layer combinations across all three faithfulness metrics. The [2,3,4] configuration shows synergistic effects, substantially outperforming the best single-layer result (layer 3).

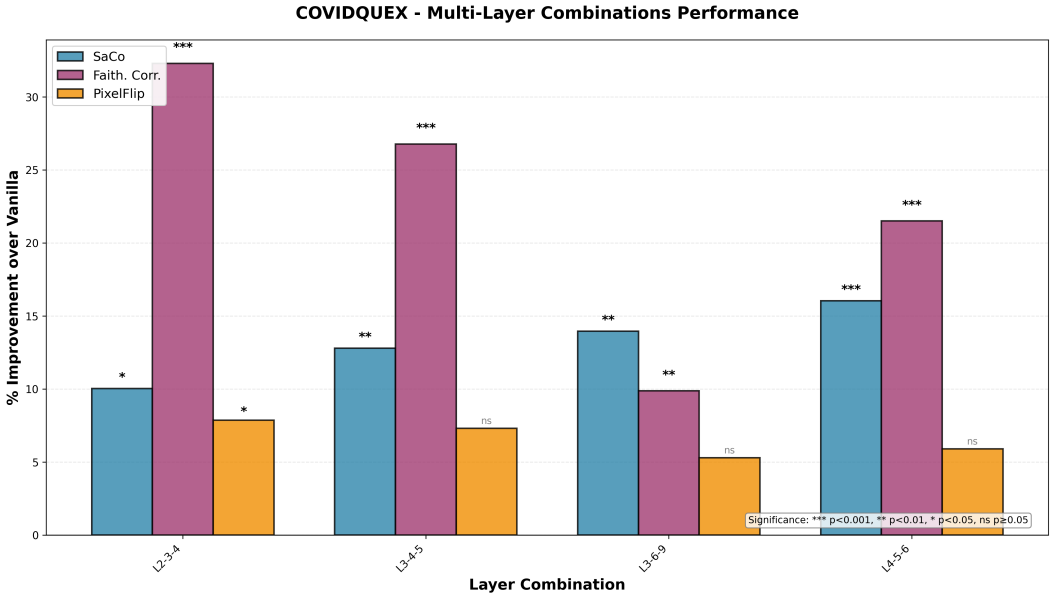


Figure 5.8: Hyperkvasir multi-layer configuration performance. The [4,6,7] configuration demonstrates strong synergistic effects across all metrics, nearly doubling the SaCo improvement compared to the best single-layer result (layer 4).

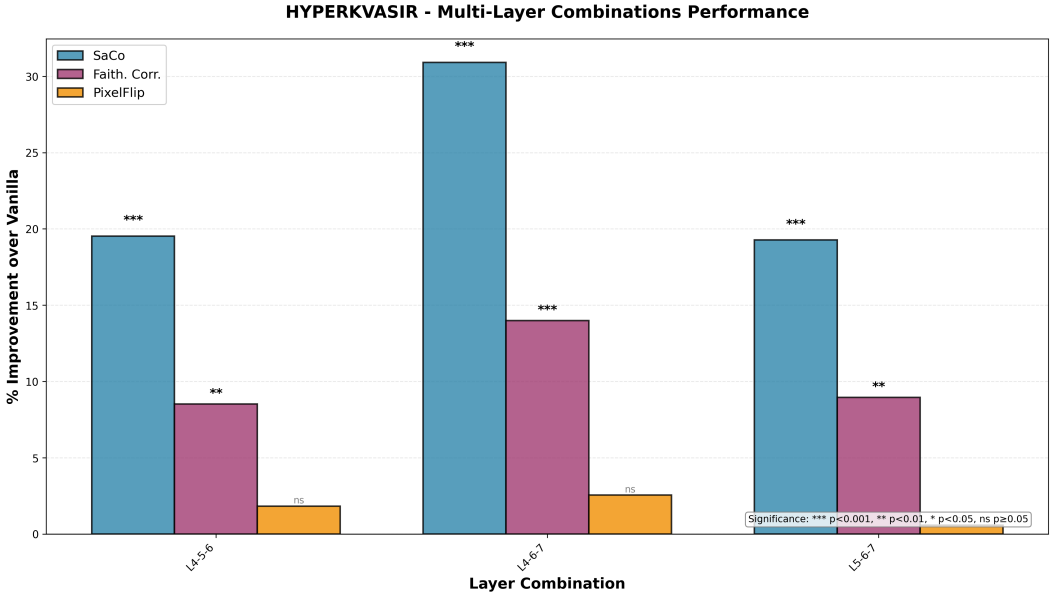
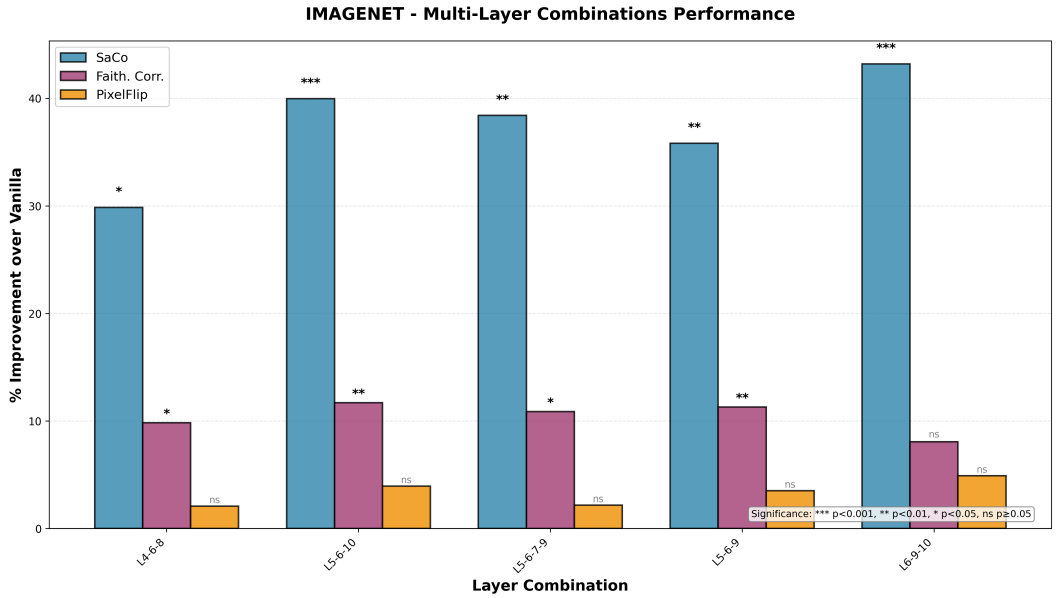


Figure 5.9: ImageNet multi-layer configuration performance. The [5,6,10] configuration shows strong synergistic effects with balanced improvements across all metrics, confirming complementary semantic information across layers.



5.3 Hyperparameter Validation

Having identified the best multi-layer configurations, we conducted systematic hyperparameter sweeps to determine optimal gate strength (κ) and maximum gate value ($c_{\max}$) for test set evaluation. For each dataset, we swept over the best-performing multi-layer combination:

- **COVID-QU-Ex:** Multi-layer configuration [2,3,4]
- **Hyperkvasir:** Multi-layer configuration [4,6,7]
- **ImageNet:** Multi-layer configuration [5,6,10]

The hyperparameter space explored: gate strength $\kappa \in \{0.1, 0.5, 1.0\}$ and maximum gate value $c_{\max} \in \{2.0, 10.0, 50.0\}$, yielding 9 combinations per dataset tested on validation sets. Figure 5.10 presents heatmaps showing percentage improvement over vanilla baseline.

5.3.1 Key Findings

Dataset-specific optimal parameters. Different datasets benefit from different hyperparameter settings, revealing task-specific sensitivity to gate strength and range:

- **COVID-QU-Ex:** Highest improvements at $\kappa = 1.0$, $c_{\max} = 50.0$ achieving 9.5% SaCo, 39.4% faithfulness correlation, and 10.9% pixel flipping improvements. Strong gate ranges ($c_{\max} = 50.0$) consistently provide substantial faithfulness correlation gains (26.6-39.4%) across all tested κ values.
- **Hyperkvasir:** Peak SaCo performance at $\kappa = 0.1$, $c_{\max} = 50.0$ with 49.2% improvement, though more moderate settings ($\kappa = 0.5$, $c_{\max} = 10.0$) achieve strong balanced performance (30.9% SaCo, 14.0% faithfulness correlation, 2.5% pixel flipping) across all metrics.
- **ImageNet:** Strongest gains at $\kappa = 0.5$, $c_{\max} = 50.0$ with 51.4% SaCo improvement and 6.2% pixel flipping. Lower $c_{\max}$ values provide better faithfulness correlation: $\kappa = 0.5$, $c_{\max} = 2.0$ achieves 9.8% improvement compared to 3.5% at $c_{\max} = 50.0$.

Robust performance regions. The combined method shows substantial positive improvements across wide regions of the hyperparameter space. All tested gate strength values (0.1-1.0) produce positive results, with higher values (0.5-1.0) generally yielding stronger improvements and $\kappa = 0.5$ providing the most consistent performance across datasets and metrics. Higher clipping ranges (50.0) maximize SaCo improvements across all datasets but show mixed effects on other metrics. Hyperkvasir and COVID-QU-Ex benefit from strong corrections ($c_{\max} = 50.0$), while ImageNet shows sensitivity with faithfulness correlation degrading at extreme values (3.5-5.5% vs. 9.8% at $c_{\max} = 2.0$).

Selected test set configurations. Based on these sweeps, we selected dataset-specific hyperparameters for test evaluation:

- **COVID-QU-Ex:** $\kappa = 1.0$, $c_{\max} = 50.0$ (maximizes all three metrics)
- **Hyperkvasir:** $\kappa = 0.1$, $c_{\max} = 10.0$ (strong SaCo gains with balanced other metrics)
- **ImageNet:** $\kappa = 0.5$, $c_{\max} = 10.0$ (balanced performance across metrics)

5.4 Test Set Evaluation

We evaluate the best validation configurations on held-out test sets. For COVID-QU-Ex and Hyperkvasir we use the full test set; for ImageNet we use a randomly selected subset of 10,000 images. Table 5.1 presents results for vanilla TransMM baseline, our method with real features, and the shuffled decoder control (methodology described in Section 4.6.5).

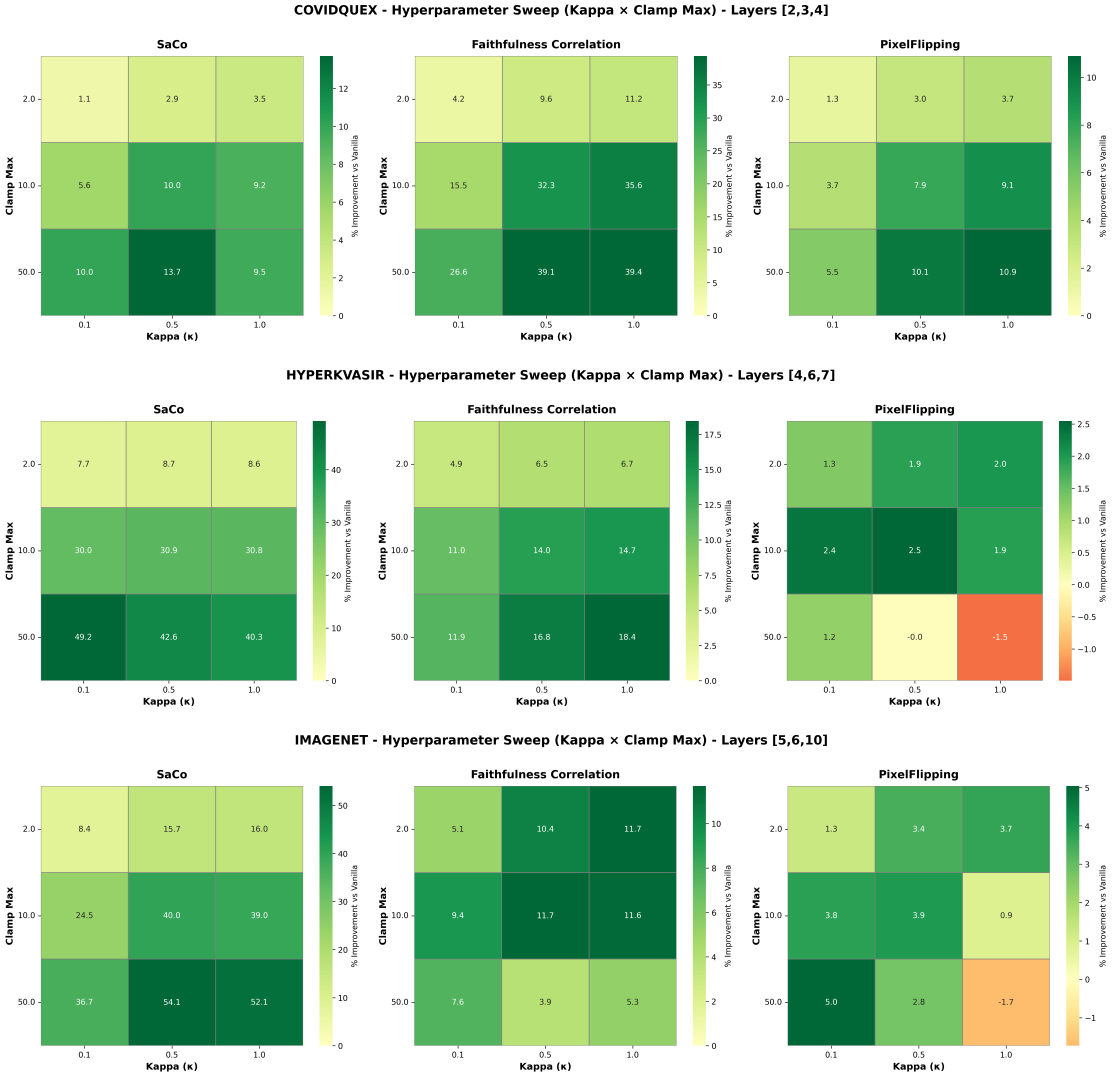
Table 5.1: Test set results comparing vanilla TransMM baseline against combined method with real and shuffled SAE features. Hyperparameters shown in Table 5.2. Bold indicates best performance per metric. Significance levels: $^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$ (two-tailed Welch's t-test vs. vanilla baseline).

Dataset	κ	$c_{\max}$
COVID-QU-Ex	1.0	50.0
Hyperkvasir	0.1	10.0
ImageNet	0.5	10.0

Table 5.2: Optimized hyperparameters

Dataset	Variant	Layers	SaCo↑	F.C.↑	Pixel↓
COVID	Vanilla	-	0.459 ± 0.004	0.309 ± 0.002	93.17 ± 0.68
	Real	[2,3,4]	$\mathbf{0.507 \pm 0.004}^{***}$	$\mathbf{0.442 \pm 0.002}^{***}$	$\mathbf{83.06 \pm 0.67}^{***}$
	Shuffled	[2,3,4]	$0.434 \pm 0.004^{***}$	$0.319 \pm 0.002^{**}$	$96.66 \pm 0.70^{***}$
Hyper.	Vanilla	-	0.504 ± 0.016	0.479 ± 0.009	94.50 ± 2.41
	Real	[4,6,7]	$\mathbf{0.656 \pm 0.015}^{***}$	$\mathbf{0.525 \pm 0.008}^{***}$	$\mathbf{92.82 \pm 2.42}$
	Shuffled	[4,6,7]	$0.567 \pm 0.017^{**}$	0.480 ± 0.008	95.44 ± 2.40
ImageNet	Vanilla	-	0.250 ± 0.004	0.314 ± 0.003	8.83 ± 0.09
	Real	[5,6,10]	$\mathbf{0.337 \pm 0.004}^{***}$	$\mathbf{0.358 \pm 0.002}^{***}$	$\mathbf{8.52 \pm 0.09}^{*}$
	Shuffled	[5,6,10]	0.242 ± 0.004	$0.250 \pm 0.003^{***}$	$9.70 \pm 0.09^{***}$

Figure 5.10: Hyperparameter sweep heatmaps showing percentage improvement over vanilla baseline. Each row represents a dataset with three metrics (SaCo, Faithfulness Correlation, Pixel Flipping). The combined method shows robust improvements across wide parameter ranges.



5.4.1 Performance Analysis

Real SAE features achieve best performance on all nine metric-dataset combinations. SaCo and Faithfulness Correlation show statistically significant improvements ($p < 0.001$) across all three datasets: COVID-QU-Ex shows 10.5% SaCo and 43.0% faithfulness correlation gains; Hyperkvasir demonstrates 30.3% SaCo and 9.7% faithfulness correlation improvements; ImageNet achieves 34.8% SaCo and 14.0% faithfulness correlation enhancements.

Pixel Flipping shows consistent improvements across all datasets with reductions of 10.8% (COVID-QU-Ex, $p < 0.001$), 1.8% (Hyperkvasir, not significant), and 3.5% (ImageNet, $p < 0.05$). The more modest Pixel Flipping improvements, while still universally positive, likely reflect this metric's documented sensitivity limitations [WKT+24b], as discussed in Chapter 6.

5.4.2 Semantic Structure Validation

The shuffled decoder control provides strong evidence that semantic structure is essential: real features consistently outperform shuffled variants by 7.3% (COVID-QU-Ex), 8.9% (Hyperkvasir), and 39.3% (ImageNet) on SaCo. Interestingly, shuffled features sometimes exceed vanilla baseline (COVID-QU-Ex faithfulness correlation: 0.319 vs. 0.309; Hyperkvasir SaCo: 0.567 vs. 0.504), suggesting sparsity provides incidental denoising benefits independent of semantics. However, full performance gains require semantic alignment, as seen on ImageNet where shuffled features actually underperform vanilla (SaCo: 0.242 vs. 0.250), likely because the 1,000-class diversity makes random feature projections actively harmful without semantic grounding.

5.5 Qualitative Attribution Analysis

To complement quantitative faithfulness metrics, we conducted qualitative inspection of attribution dynamics on ImageNet. We isolated the top 100 images showing improvement across a composite of all three metrics and identified patches with largest attribution magnitude changes ($|\Delta\text{Attribution}|$) between baseline and gated models. For each patch, we extracted the SAE feature contributing most to the change.

While individual SAE feature interpretability remains challenging, with many features exhibiting polysemanticity or abstract activations, our analysis revealed distinct behavioral patterns. By examining prototypical activations (top-10 activating validation images) for these features, we identify strategies the gating mechanism employs to refine attributions. We categorize observed behaviors into success patterns (Section 5.5.1) where semantic steering improves faithfulness, and failure modes (Section 5.5.2) where over-aggressive corrections degrade performance.

5.5.1 Success Patterns

Suppression of confounding concepts. A recurring pattern is active suppression of high-salience concepts that co-occur with target classes but lack causal relevance. Features L6-10968 and L9-45529 were identified as facial detectors, activating on eyes and nose regions in prototypes (Figures 5.11 and 5.12). In case studies, these features consistently appeared with negative attribution contributions (suppressors) in classes such as *Academic Gown*, *Cloak*, or *Golden Retriever*.

While faces are visually prominent, they are confounders for clothing classification. The gating mechanism utilizes gradients of these features to down-weight attention on persons, redirecting focus toward relevant objects. This aligns with negative semantic steering, where the model explicitly identifies and subtracts irrelevant concepts. Notably, Layer 6 features do not distinguish between animal and human faces, while Layer 9 features specifically target human facial structures.

Mitigation of dataset artifacts. Feature-gradient mechanisms aid in filtering non-semantic data artifacts. Feature L6-37778 activates strongly on text overlays and watermarks (Figure 5.13). In analyzed samples, this feature suppressed patches containing copyright labels, probably preventing the model from exploiting metadata text as prediction shortcuts.

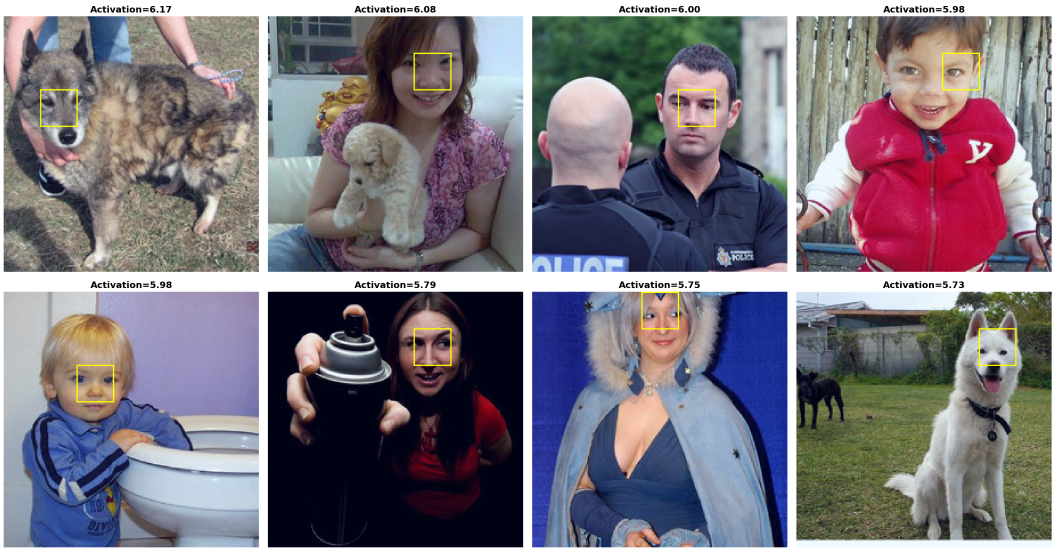
Similarly, Feature L9-6561 detects specific red textures (Figure 5.14). Its suppression in classes like *Hamper* or *Perfume* suggests removal of background color biases (e.g., red blankets) that correlate with but do not cause object classes. In classes like *Bell Pepper* or *Flamingo*, it de-emphasizes regions providing minimal additional context.

Context-dependent feature modulation. Certain features exhibit sophisticated context-switching behavior, acting as boosters or suppressors depending on semantic context. Feature L6-20254 detects animal fur textures (Figure 5.15). In images of *Schipperke* or *Border Collies*, where fur defines the class, this feature boosts attribution. Conversely, in images of *Pekinese*, the same feature suppresses attribution.

This modulation suggests the gating mechanism is not a static filter but adapts to context. In suppression cases, the model may dampen generic fur textures to resolve feature crowding, forcing attribution to rely on more discriminative features such as facial geometry to distinguish similar dog breeds.

Broadband noise filtering. A class of features (e.g., Feature L10-5436) appears across diverse unrelated classes (Figure 5.16). Prototypes display high-frequency edges, blur, or generic gradients. Their ubiquity and consistent suppressive role suggest they function as broadband background filters. By suppressing low-information patches, the gating mechanism might perform late-stage denoising, cleaning attribution maps of visual clutter.

(a) Case studies: Suppression of faces in clothing classes.

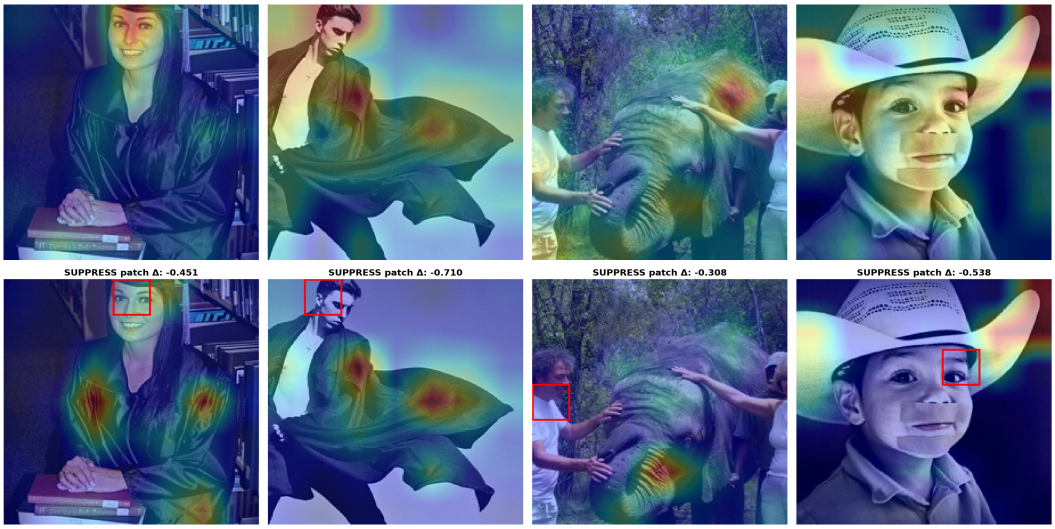


(b) Prototypes: Top activating images for Feature L6-10968 (facial detector).

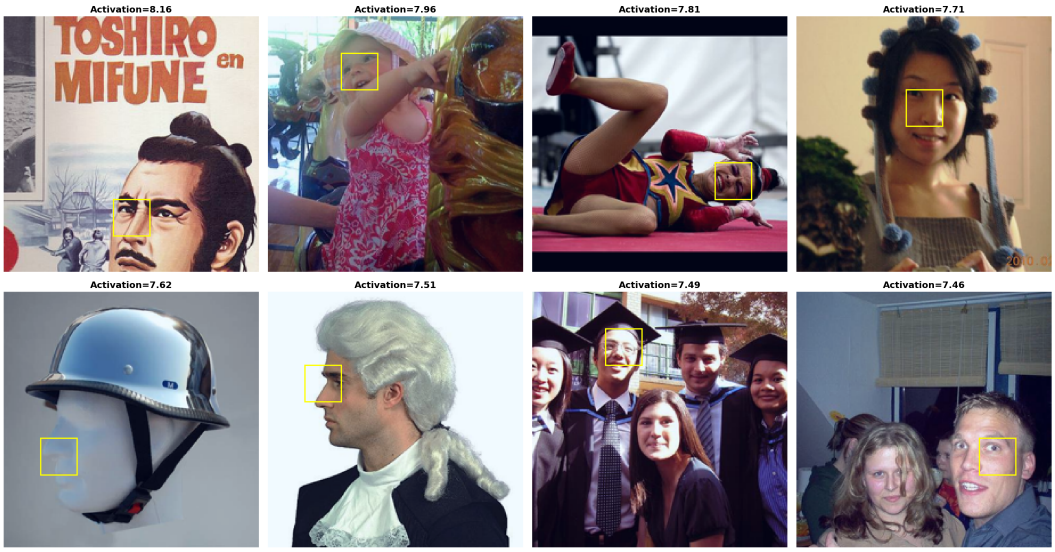
Figure 5.11: Feature L6-10968 detects facial landmarks (bottom) and is actively suppressed (red boxes, top) in classes where the person is secondary to the object (e.g., Academic Gown).

5.5.2 Failure Analysis

To understand method limitations, we analyzed samples where gating degraded faithfulness metrics. Suppression strategies that improve performance in some contexts cause degradation when applied over-aggressively.



(a) Case studies: Semantic suppression in deeper layers.



(b) Prototypes: Feature L9-45529 (human face context).

Figure 5.12: Feature L9-45529 exhibits similar facial suppression behavior in deeper layers, refining object classification by removing human presence from attribution.

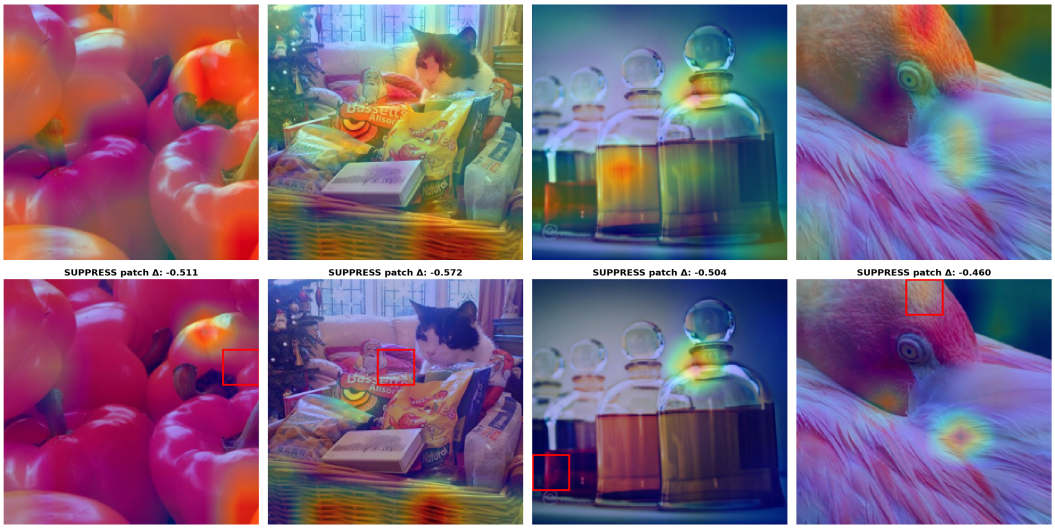
Contextual misalignment in feature suppression. A notable failure mode involves Feature L6-20254, the context-switching feature that typically boosts fur attribution for dogs (Figure 5.15). However, in failure cases such as *Giant Panda* (Figure 5.17), the gating mechanism incorrectly identifies defining fur texture as background noise and suppresses it, removing class-defining signals.

(a) Case studies: Suppression of text artifacts.

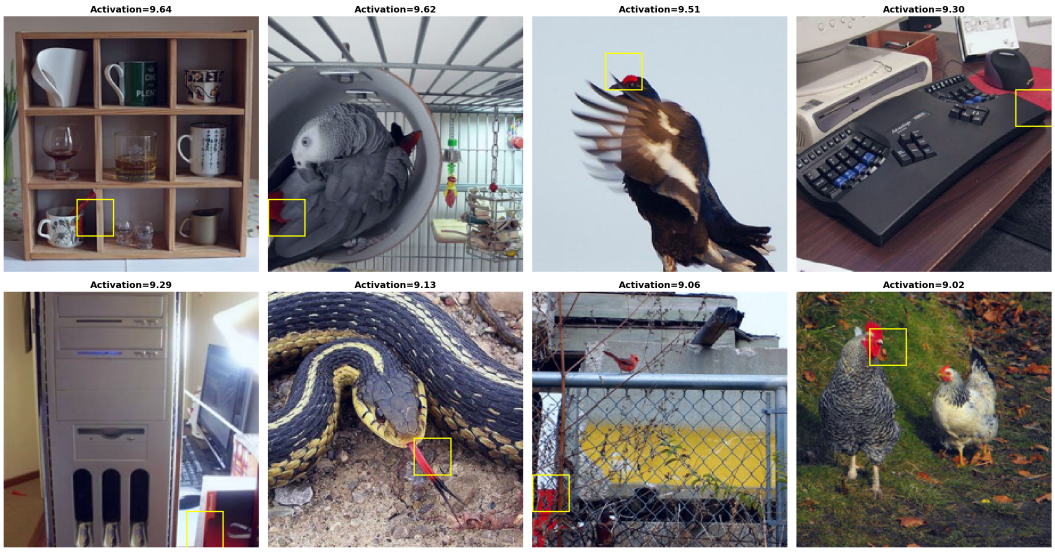
(b) Prototypes: Feature L6-37778 (text/watermark detector).

Figure 5.13: Feature L6-37778 identifies artificial text overlays (bottom) and suppresses them (top) to prevent spurious predictions based on watermarks.

Excessive suppression of class-defining features. Feature L6-8584 was the most frequent cause of degradation in Layer 6. Analysis of prototypes (Figure 5.18b) reveals it detects feathers and wildlife skin textures. While this feature likely aids in separating birds from complex backgrounds, it acts almost exclusively as a suppressor (96% of cases). In failure cases (e.g., *Vulture*, *Bald Eagle*), suppression was applied to the object itself, erasing subject texture and leaving feature-less silhouettes (Figure 5.18a).

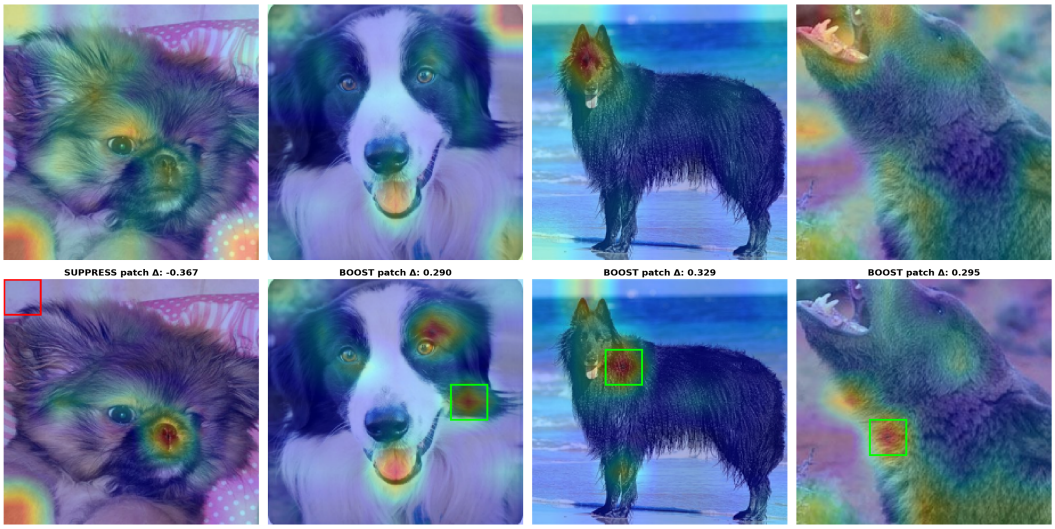


(a) Case studies: Suppression of background color bias.

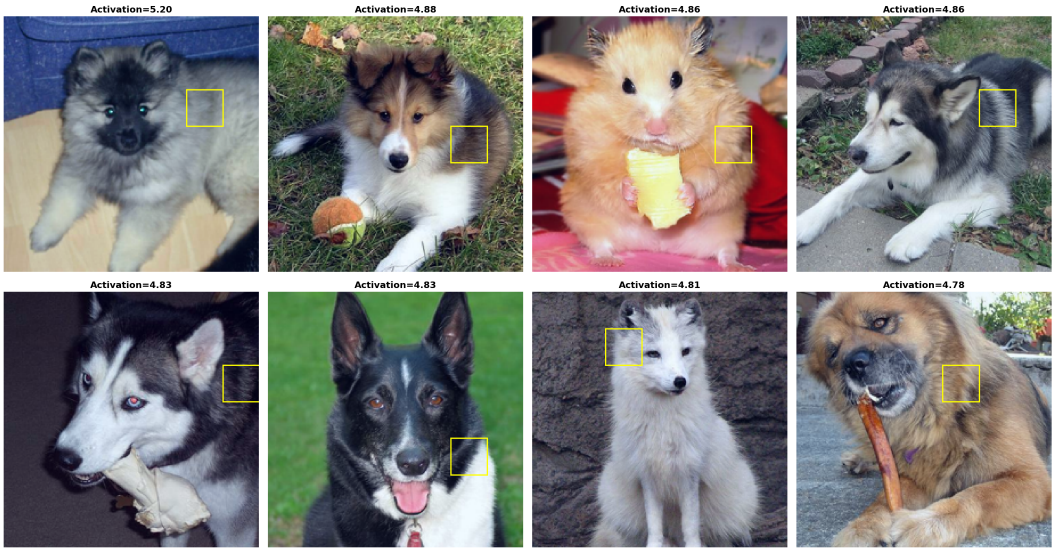


(b) Prototypes: Feature L9-6561 (red texture detector).

Figure 5.14: Feature L9-6561 suppresses red background textures (e.g., velvet) often correlated with specific object categories, removing spurious color biases.

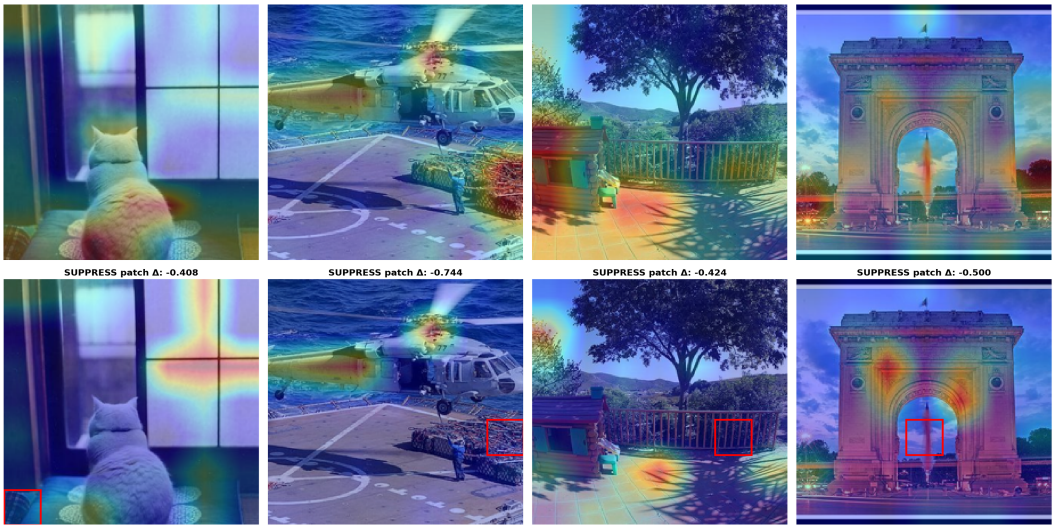


(a) Case studies: Context-dependent switching behavior.



(b) Prototypes: Feature L6-20254 (fur texture detector).

Figure 5.15: Feature L6-20254 exhibits context-dependent behavior, boosting attribution (green boxes) when fur defines the object but suppressing it (red boxes) when more discriminative features are needed.



(a) Case studies: Background noise removal.

(b) Prototypes: Feature L10-5436 (broadband noise detector).

Figure 5.16: Feature L10-5436 detects generic background noise/blur (bottom) and suppresses it across diverse classes (top), providing consistent denoising.

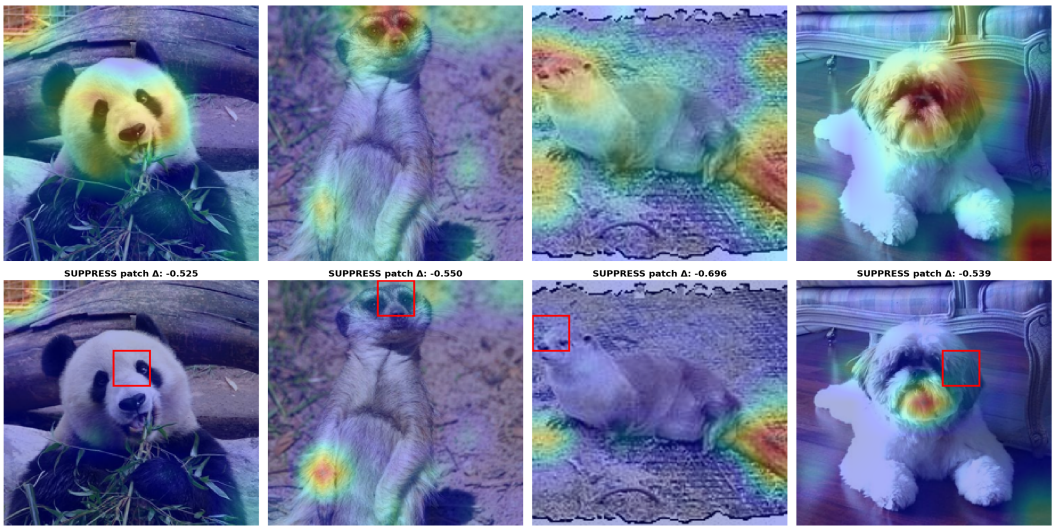
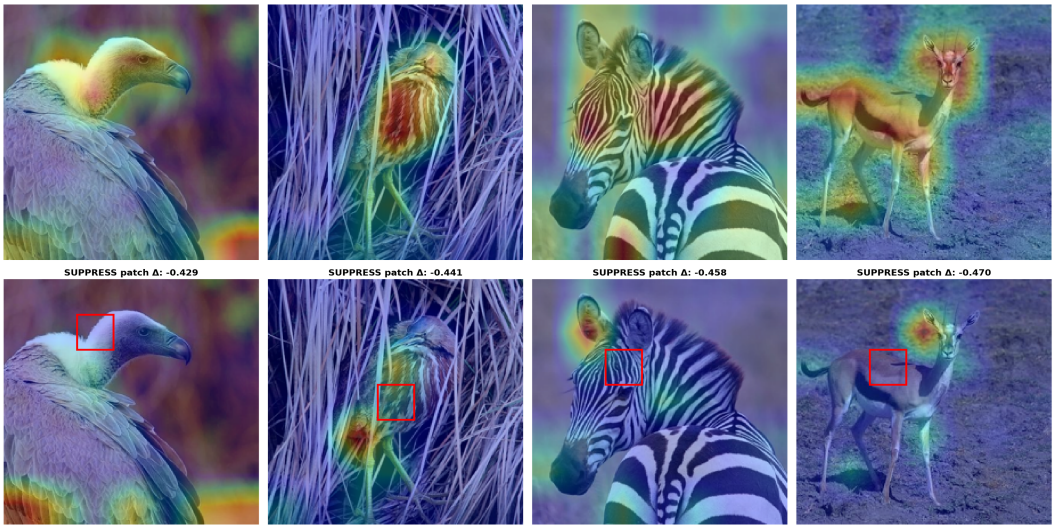
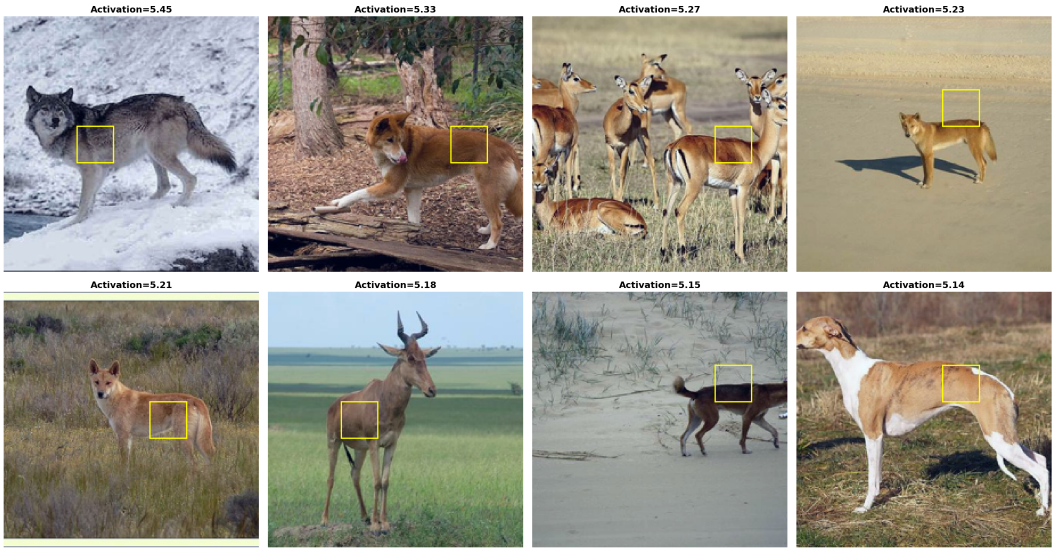


Figure 5.17: Contextual misalignment by Feature L6-20254. Unlike success cases in Figure 5.15, here the feature suppresses Giant Panda fur texture (red boxes), removing vital class-defining information.



(a) Case studies: Excessive suppression of defining textures.



(b) Prototypes: Feature L6-8584 (feather/wildlife texture detector).

Figure 5.18: Feature L6-8584 detects wildlife textures/feathers (bottom) but in failure cases suppresses the object itself (top, e.g., Vulture), degrading attribution by removing class-defining features.

CHAPTER 6

Discussion and Conclusion

6.1 The Necessity of Semantic Grounding

TransMM's success relies on a core principle: combining attention patterns (which patches receive how much attention) with gradients (how that attention affects predictions) to identify important regions. Our method tests whether this same principle extends to sparse autoencoder features, where activations indicate which semantic concepts are present and gradients indicate how those concepts influence classification.

Our ablation studies provide evidence that this extension holds. The combined method of using both feature activations and gradients consistently outperforms vanilla TransMM across all datasets and metrics. Critically, neither signal alone achieves comparable results. The activation-only variant shows modest improvements, suggesting that knowing which features are present provides some attribution signal, but it lacks the gradient information to distinguish merely present features from causally influential ones. Conversely, the gradient-only variant performs at or below baseline, suggesting that gradient information without activation context can mislead attribution.

This raises a critical question: do these improvements stem from the semantic correspondence between SAE features and visual concepts, or merely from statistical properties of sparse decomposition? Our shuffled decoder control addresses this by preserving all statistical properties, like sparsity patterns and activation magnitudes, while destroying the correspondence between feature dimensions and semantic concepts.

While shuffled features sometimes exceed the vanilla baseline, suggesting sparsity provides incidental benefits such as noise reduction, real SAE features consistently and substantially outperform both. This provides strong evidence that improvements require **semantic alignment**, not just sparse decomposition. The faithful attribution requires both components of our proposed principle: the semantic "mask" of activation (knowing what is present) and the causal direction of the gradient (knowing what matters).

6.2 Feature Locality and Multi-Layer Synergy

Our empirical finding that intermediate layers consistently provide optimal attribution improvements (layers 4–7 for medical datasets) may relate to feature locality properties documented in recent SAE research. Han et al. [HKK25] found that feature non-locality remains minimal through the early network but becomes significant in later layers.

In our models, the optimal intermediate range corresponds to a potential balance point. Early layers (1–3) show minimal gains, suggesting their low-level features lack sufficient semantic content for classification-relevant attribution. Later layers (8–10) show variable performance, potentially due to increasing feature non-locality as self-attention causes spatial mixing. The intermediate layers may thus represent a region where features have developed semantic richness while maintaining the spatial coherence useful for patch-level attribution.

This hierarchical understanding helps explain the performance of our multi-layer configurations. The substantial gains observed when combining features from adjacent layers suggest that features provide synergistic information beyond simple additive effects.

Two non-exclusive mechanisms could explain this finding. First, features representing similar semantic concepts may activate across consecutive layers, providing independent evidence that reinforces important patches. Second, adjacent layers may capture complementary aspects such as textures, object parts, and global semantics to form a more complete representation. Most likely, both mechanisms contribute: patches containing consistent evidence across multiple semantic levels receive stronger amplification than those with evidence at only one level.

6.3 Implications for Medical Imaging

Our motivation for this work stemmed from the need for trustworthy AI in medical imaging, where clinicians must understand not just what a model predicts but why. The faithfulness improvements demonstrated on medical datasets (COVID-QU-Ex, Hyper-kvasir) suggest that SAE-enhanced attribution could help verify that models rely on clinically meaningful features rather than spurious correlations.

However, translating improved faithfulness metrics into clinical trust requires additional validation. Future work should investigate whether doctors find SAE-enhanced explanations more useful for detecting problematic model behavior, and whether the semantic interpretability of SAE features (e.g., "this patch activated features representing lung infiltrates") provides actionable insights beyond pixel-level heatmaps. The computational overhead of SAE training and multi-layer inference may also limit deployment in resource-constrained clinical settings, suggesting that single-layer configurations or pre-trained SAEs may be more practical for real-world adoption.

6.4 Limitations

Computational Overhead Our method requires training SAEs on model activations before attribution can be computed, adding a preprocessing step not required by vanilla TransMM. The computational cost of SAE training depends on dataset size and the number of layers for which SAEs are needed. Larger datasets and more layers naturally require proportionally more compute. Once trained, multi-layer configurations require loading multiple SAEs simultaneously (e.g., three SAEs for layers [4,5,6]), which increases memory requirements compared to single-layer or vanilla approaches. Inference with multi-layer gating scales linearly with the number of layers used, as each layer requires SAE encoding and gate computation.

Varied Magnitudes Across Metrics While our method improves all three faithfulness metrics across all datasets, the magnitude of improvement varies: SaCo and Faithfulness Correlation show substantial gains (10–43%), while Pixel Flipping shows more modest but consistent improvements (2–11%). This pattern holds across all three datasets, suggesting our method particularly enhances magnitude alignment and correlation aspects of faithfulness captured by SaCo and Faithfulness Correlation.

We hypothesize that this differential stems from how the metrics weight different regions of the attribution distribution. Pixel Flipping, by design, heavily prioritizes the highest-ranked patches, removing them first and measuring immediate impact. If TransMM already identifies the most critical regions reasonably well, there is limited room for improvement in this top-ranked subset. In contrast, both SaCo and Faithfulness Correlation evaluate the full attribution distribution more evenly: SaCo tests all pairwise comparisons between patch groups, while Faithfulness Correlation samples random subsets across the entire range. Our method's primary contribution appears to be refining attributions in moderately-important regions, de-emphasizing patches that receive moderate attribution scores but are ultimately irrelevant while preserving or strengthening truly important regions. These refinements have substantial impact on metrics that evaluate the complete distribution (SaCo, Faithfulness Correlation) but smaller impact on metrics that focus predominantly on the top-ranked patches (Pixel Flipping). This interpretation is consistent with Pixel Flipping's documented sensitivity limitations [WKT+24b], particularly its reduced sensitivity to changes outside the highest-attribution regions.

6.5 Future Work

Understanding Where Improvements Occur The modest Pixel Flipping improvements despite substantial SaCo gains suggest our method may refine moderately important patches rather than changing the most obvious regions. Future work should analyze the distribution of attribution changes: do gates primarily affect high-attribution patches (refining already-identified regions) or moderate-attribution patches (discovering additional relevant regions)? Computing the distribution of gate values and correlating with

attribution magnitudes could reveal whether the method provides contextual refinement or discovers new important regions.

Comparison with Complementary Methods TokenTM [WKT+24a] improves TransMM by incorporating MLP transformation information. Comparing our SAE-based feature gating with TokenTM's approach could reveal whether these methods capture complementary aspects of model computation, and whether they could be combined for further improvements.

Extension to Other Attribution Methods Our approach demonstrates that sparse semantic features can enhance gradient-based attribution. Future work should investigate whether similar principles apply to other attribution frameworks. Could SAE features improve GradCAM, Integrated Gradients, or attention rollout methods? This would test whether feature-gradient decomposition is a general principle or specific to TransMM's architecture.

Application to Other Vision Tasks The success of semantic feature gating for classification attribution suggests potential applications to other vision tasks requiring spatial explanations, such as object detection, semantic segmentation, or visual question answering. Whether the same principles extend to these tasks, where attribution targets are more complex than single class labels, remains an open question.

6.6 Conclusion

This thesis investigated whether incorporating semantic structure from Sparse Autoencoders could improve attribution faithfulness in Vision Transformers. Building on TransMM's principle that faithful attribution requires combining activation patterns with gradient information, we demonstrated that this principle extends naturally to learned sparse features: feature activations indicate which semantic concepts are present, while gradients indicate how those concepts influence predictions.

Through comprehensive evaluation across three datasets, two model architectures, and three complementary faithfulness metrics, we demonstrated that feature-gradient gating improves attribution quality across all metrics without exception. Real SAE features achieve 10.5–34.8% improvements on SaCo, 9.7–43.0% on Faithfulness Correlation, and 1.8–10.8% on Pixel Flipping compared to vanilla TransMM, with statistical significance achieved on all metrics for one dataset and on SaCo and Faithfulness Correlation for all three datasets. Ablation studies confirmed that both signals are necessary and that neither activation-only nor gradient-only variants match the combined method's performance. The shuffled decoder control demonstrated that improvements require semantic alignment between features and visual concepts, not merely statistical properties of sparse decomposition.

Our results suggest that mechanistic interpretability tools, developed primarily for understanding model internals, can directly enhance practical explainability methods. The success of this approach across medical imaging and natural images, fine-tuned and contrastively pre-trained models, indicates broad applicability. This work provides evidence that attribution methods should consider leveraging learned semantic representations that respect the structure of model computations, opening new directions for faithful visual explanations in Vision Transformers.

List of Figures

List of Tables

Bibliography

[AB16] Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016.

[AGM+18] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. *Advances in neural information processing systems*, 31, 2018.

[AHD+24] Reduan Achtibat, Sayed Mohammad Vakilzadeh Hatefi, Maximilian Dreyer, Aakriti Jain, Thomas Wiegand, Sebastian Lapuschkin, and Wojciech Samek. AttnLRP: Attention-aware layer-wise relevance propagation for transformers. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 135–168. PMLR, 21–27 Jul 2024.

[Ano25] Anonymous. Steering CLIP's vision transformer with sparse autoencoders. In *Mechanistic Interpretability for Vision at CVPR 2025 (Non-proceedings Track)*, 2025.

[AZ20] Samira Abnar and Willem H. Zuidema. Quantifying attention flow in transformers. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 4190–4197. Association for Computational Linguistics, 2020.

[BKH16] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[BTB+23] Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E Burke, Tristan Hume, Shan Carter,

Tom Henighan, and Christopher Olah. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023.

[BTS+20] Hanna Borgli, Vajira Thambawita, Pia H Smedsrud, Steven Hicks, Debesh Jha, Sigrun L Eskeland, Kristin Ranheim Randel, Konstantin Pogorelov, Mathias Lux, Duc Tien Dang Nguyen, Dag Johansen, Carsten Griwodz, Håkon K Stensland, Enrique Garcia-Ceja, Peter T Schmidt, Hugo L Hammer, Michael A Riegler, Pål Halvorsen, and Thomas de Lange. HyperKvasir, a comprehensive multi-class image and video dataset for gastrointestinal endoscopy. *Scientific Data*, 7(1):283, 2020.

[BWM20] Umang Bhatt, Adrian Weller, and José M. F. Moura. Evaluating and aggregating feature-based model explanations. In Christian Bessiere, editor, *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, pages 3016–3022. ijcai.org, 2020.

[CGC+20] Nick Cammarata, Gabriel Goh, Shan Carter, Ludwig Schubert, Michael Petrov, and Chris Olah. Curve detectors. *Distill*, 2020.

[CGW21a] Hila Chefer, Shir Gur, and Lior Wolf. Generic attention-model explainability for interpreting bi-modal and encoder-decoder transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 397–406, October 2021.

[CGW21b] Hila Chefer, Shir Gur, and Lior Wolf. Transformer interpretability beyond attention visualization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 782–791, June 2021.

[CJJ+24] Noel C. F. Codella, Ying Jin, Shrey Jain, Yu Gu, Ho Hin Lee, Asma Ben Abacha, Alberto Santamaria-Pang, Will Guyman, Naiteek Sangani, Sheng Zhang, Hoifung Poon, Stephanie Hyland, Shruthi Bannur, Javier Alvarez-Valle, Xue Li, John Garrett, Alan McMillan, Gaurav Rajguru, Madhu Maddi, Nilesh Vijayrania, Rehaan Bhimai, Nick Mecklenburg, Rupal Jain, Daniel Holstein, Naveen Gaur, Vijay Aski, Jenq-Neng Hwang, Thomas Lin, Ivan Tarapov, Matthew Lungren, and Mu Wei. Medimageinsight: An open-source embedding model for general domain medical imaging, 2024.

[CMPL+23] Arthur Conmy, Augustine Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. Towards automated circuit discovery for mechanistic interpretability. *Advances in Neural Information Processing Systems*, 36:16318–16352, 2023.

[DBK+21] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer,

Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

[DCLT19] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *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, June 2019. Association for Computational Linguistics.

[EHO+22] Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. Toy models of superposition. *Transformer Circuits Thread*, 2022. `https://transformer-circuits.pub/2022/toy_model/index.html`.

[ENO+21] Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021.

[GDlTT+25] Leo Gao, Tom Dupre la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu, editors, *International Conference on Representation Learning*, volume 2025, pages 26721–26754, 2025.

[GIF+23] Samir Yitzhak Gadre, Gabriel Ilharco, Alex Fang, Jonathan Hayase, Georgios Smyrnis, Thao Nguyen, Ryan Marten, Mitchell Wortsman, Dhruba Ghosh, Jieyu Zhang, Eyal Orgad, Rahim Entezari, Giannis Daras, Sarah Pratt, Vivek Ramanujan, Yonatan Bitton, Kalyani Marathe, Stephen Mussmann, Richard Vencu, Mehdi Cherti, Ranjay Krishna, Pang Wei Koh, Olga Saukh, Alexander Ratner, Shuran Song, Hannaneh Hajishirzi, Ali Farhadi, Romain Beaumont, Sewoong Oh, Alex Dimakis, Jenia Jitsev, Yair Carmon, Vaishaal Shankar, and Ludwig Schmidt. Datacomp: In search of the next generation of multimodal datasets. *arXiv preprint arXiv:2304.14108*, 2023.

[HCS+24] Robert Huben, Hoagy Cunningham, Logan Riggs Smith, Aidan Ewart, and Lee Sharkey. Sparse autoencoders find highly interpretable features

in language models. In *The Twelfth International Conference on Learning Representations*, 2024.

[HG17] Dan Hendrycks and Kevin Gimpel. Bridging nonlinearities and stochastic regularizers with gaussian error linear units, 2017.

[HKK25] Sangyu Han, Yearim Kim, and Nojun Kwak. Causal interpretation of sparse autoencoder features in vision, 2025.

[HZRS16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society, 2016.

[JEP+21] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *nature*, 596(7873):583–589, 2021.

[JG20] Alon Jacovi and Yoav Goldberg. Towards faithfully interpretable NLP systems: How should we define and evaluate faithfulness? In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4198–4205, Online, July 2020. Association for Computational Linguistics.

[JSH+25] Sonia Joseph, Praneet Suresh, Lorenz Hufe, Edward Stevinson, Robert Graham, Yash Vadi, Danilo Bzdok, Sebastian Lapuschkin, Lee Sharkey, and Blake Aaron Richards. Prisma: An open source toolkit for mechanistic interpretability in vision and video, 2025.

[JW19] Sarthak Jain and Byron C. Wallace. Attention is not Explanation. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *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, June 2019. Association for Computational Linguistics.

[KB15] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[KBB23] Piotr Komorowski, Hubert Baniecki, and Przemyslaw Biecek. Towards evaluating explanations of vision transformers for medical imaging. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 3726–3732, June 2023.

[KK01] John F. Kolen and Stefan C. Kremer. *Gradient Flow in Recurrent Nets: The Difficulty of Learning LongTerm Dependencies*, pages 237–243. Wiley-IEEE Press, 2001.

[KME+24] Maxime Kayser, Bayar Menzat, Cornelius Emde, Bogdan Bercean, Bartlomiej W. Papiez, Alex Novak, Abdala Espinosa, Susanne Gaube, Thomas Lukasiewicz, and Oana-Maria Camburu. Fool me once? contrasting vision- and language-based explanations in a clinical decision-support setting. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, Florida, November 12-16, 2024*, November 2024.

[LBBH98] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[LBM+15] Sebastian Lapuschkin, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS ONE*, 10:e0130140, 07 2015.

[LCCS25] Hyesu Lim, Jinho Choi, Jaegul Choo, and Steffen Schneider. Sparse autoencoders reveal selective remapping of visual concepts during adaptation. In *The Thirteenth International Conference on Learning Representations*, 2025.

[MBSP21] A. K. Mondal, A. Bhattacharjee, P. Singla, and A. P. Prathosh. xViTCOS: Explainable vision transformer based COVID-19 screening using radiography. *IEEE Journal of Translational Engineering in Health and Medicine*, 10:1100110, 2021. PMID: 34956741; PMCID: PMC8691725.

[MCCD13] Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In Yoshua Bengio and Yann LeCun, editors, *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*, 2013.

[MLB+17] Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern recognition*, 65:211–222, 2017.

[OCS+20] Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. Zoom in: An introduction to circuits. *Distill*, 2020.

[OMS17] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization. *Distill*, 2017.

[PGD+20] Danish Pruthi, Mansi Gupta, Bhuwan Dhingra, Graham Neubig, and Zachary C. Lipton. Learning to deceive with attention-based explanations. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4782–4793, Online, July 2020. Association for Computational Linguistics.

[RDS+15] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[RKH+21] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 2021.

[RNSS18] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training, 2018.

[RUK+21] Maithra Raghu, Thomas Unterthiner, Simon Kornblith, Chiyuan Zhang, and Alexey Dosovitskiy. Do vision transformers see like convolutional neural networks? *Advances in neural information processing systems*, 34:12116–12128, 2021.

[SBB22] Lee Sharkey, Dan Braun, and Beren. Interim research report: Taking features out of superposition with sparse autoencoders. Alignment Forum, 12 2022.

[SBM+16] Wojciech Samek, Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, and Klaus-Robert Müller. Evaluating the visualization of what a deep neural network has learned. *IEEE transactions on neural networks and learning systems*, 28(11):2660–2673, 2016.

[SCB+25] Lee Sharkey, Bilal Chughtai, Joshua Batson, Jack Lindsey, Jeffrey Wu, Lucius Bushnaq, Nicholas Goldowsky-Dill, Stefan Heimersheim, Alejandro Ortega, Joseph Isaac Bloom, Stella Biderman, Adrià Garriga-Alonso, Arthur Conmy, Neel Nanda, Jessica Mary Rumbelow, Martin Wattenberg, Nandi Schoots, Joseph Miller, William Saunders, Eric J Michaud, Stephen

Casper, Max Tegmark, David Bau, Eric Todd, Atticus Geiger, Mor Geva, Jesse Hoogland, Daniel Murfet, and Thomas McGrath. Open problems in mechanistic interpretability. *Transactions on Machine Learning Research*, 2025. Survey Certification.

[SCBWS25] Samuel Stevens, Wei-Lun Chao, Tanya Berger-Wolf, and Yu Su. Sparse autoencoders for scientifically rigorous interpretation of vision models, 2025.

[SCD+17] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 618–626, 2017.

[SM24] Edward Sanderson and Bogdan J Matuszewski. A study on self-supervised pretraining for vision problems in gastrointestinal endoscopy. *IEEE Access*, 12:46181–46201, 2024.

[TCK+21] Anas M. Tahir, Muhammad E.H. Chowdhury, Amith Khandakar, Tawsifur Rahman, Yazan Qiblawey, Uzair Khurshid, Serkan Kiranyaz, Nabil Ibtehaz, M. Sohel Rahman, Somaya Al-Maadeed, Sakib Mahmud, Maymouna Ezeddin, Khaled Hameed, and Tahir Hamid. Covid-19 infection localization and severity grading from chest x-ray images. *Computers in Biology and Medicine*, 139:105002, 2021.

[TCM+24] Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, Hoagy Cunningham, Nicholas L Turner, Callum McDougall, Monte MacDiarmid, C. Daniel Freeman, Theodore R. Sumers, Edward Rees, Joshua Batson, Adam Jermyn, Shan Carter, Chris Olah, and Tom Henighan. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. *Transformer Circuits Thread*, 2024.

[VSP+17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008, 2017.

[WKT+24a] Junyi Wu, Weitai Kang, Hao Tang, Yuan Hong, and Yan Yan. On the faithfulness of vision transformer explanations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10936–10945, June 2024.

[WKT+24b] Junyi Wu, Weitai Kang, Hao Tang, Yuan Hong, and Yan Yan. On the faithfulness of vision transformer explanations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10936–10945, 2024.

[WVC+23] Kevin Ro Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: a circuit for indirect object identification in GPT-2 small. In *The Eleventh International Conference on Learning Representations*, 2023.

[WW22] Yipei Wang and Xiaoqian Wang. A unified study of machine learning explanation evaluation metrics. *arXiv preprint arXiv:2203.14265*, 2022.

[YCN+15] Jason Yosinski, Jeff Clune, Anh Mai Nguyen, Thomas J. Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. *CoRR*, abs/1506.06579, 2015.

[ZF14] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.