Mechanistic Interpretability of Transformer MLPs Through Exact Soft-Gate Decomposition

Anonymous authors
Paper under double-blind review

Abstract

Transformer feed-forward (MLP) blocks perform nonlinear transformations, but their internal structure is often difficult to interpret using current methods. This paper presents an exact soft-gate decomposition that extracts the effective weight matrix for each MLP block at varying inputs. This approach generalises exact piecewise-affine decomposition from ReLU networks to Gaussian Error Linear Unit (GELU) and Sigmoid Linear Unit (SiLU) activations, covering both standard and gated MLP architectures. Verification across five models, including BERT, mBART, and GPT-2 for language tasks, ViT for vision tasks, and the gated SiLU architecture TinyLlama, confirms the accuracy of this method at floating-point arithmetic precision. The application of singular value decomposition to the effective weight matrix yields six structural metrics that characterise each block based on the dimensionality, intensity, and spectral concentration of its transformation. When applied to BERT, mBART, and GPT-2, and alongside ablation studies that provide causal ground truth, this analysis indicates that MLP computation is organised differently across encoder-only, encoder-decoder, and decoder-only models. Specifically, BERT blocks vary by transformation dimensionality, mBART blocks by transformation intensity, and GPT-2 blocks by intensity, which shows detrimental excess in the later layers. These structural differences, revealed only through the exact effective weight matrix, explain why scalar importance measures often fail to generalise across different architectures.

1 Introduction

The feed-forward (MLP) blocks in transformer architectures (Vaswani et al., 2017) constitute approximately two-thirds of the model's parameters and perform a nonlinear transformation at every layer. Despite their prominence, the internal structure of these transformations remains largely opaque. Existing interpretability methods examine transformer components through various approaches, including representations (probing), sensitivity analysis (gradient-based attribution), causal effects (ablation and activation patching), and neuron-level circuit tracing (Olah et al., 2020; Elhage et al., 2021). While these techniques have yielded valuable insights into what transformers encode and which components matter, none have been able to extract the exact transformation that an MLP block applies to a given input. As a result, the mathematical nature of the transformation itself remains inaccessible.

This gap is significant. Without access to the transformation structure, it is impossible to determine whether an MLP block achieves its effect by operating across many independent directions or by amplifying a few strongly. Additionally, it is not possible to ascertain whether the block preserves input features or mixes them, nor to compare its spectral profile with that of other blocks in the same model. These structural properties are crucial for mechanistic understanding, as they define *how* a block computes rather than just *what* it produces or *how much* it matters.

The present work addresses this gap through an exact soft-gate decomposition of transformer MLP blocks. Building on ReLU Region Reasoning (Re3) (Barua et al., 2026), which introduced piecewise-affine decomposition for ReLU networks using binary activation masks, the framework replaces the binary gate with a continuous soft gate derived from the multiplicative structure of the Gaussian Error Linear Unit (GELU),

defined as ($\phi(z) = z \cdot \Phi(z)$) and the Sigmoid Linear Unit (SiLU), defined as ($\phi(z) = z \cdot \sigma(z)$) activations. For any input $\mathbf{x}$, the MLP output is decomposed exactly as $\mathrm{MLP}(\mathbf{x}) = \mathbf{A}_x\mathbf{x} + \mathbf{D}_x$, where $\mathbf{A}_x \in \mathbb{R}^{d\times d}$ is the effective weight matrix encoding the complete linear transformation at that input. The decomposition extends to gated MLP architectures with three weight matrices, such as LLaMA (Touvron et al., 2023) and Mistral (Jiang et al., 2023), yielding $\mathrm{MLP}(\mathbf{x}) = \mathbf{A}_x\mathbf{x}$ with zero approximation error.

The effective weight matrix $\mathbf{A}_x$ is then analysed using singular value decomposition, yielding six structural metrics, and they are the Frobenius norm, the spectral norm, the nuclear norm, the effective rank, the stable rank, and the trace. These metrics characterise the dimensionality, intensity, and spectral concentration of each block's transformation. This analysis, applied to three transformer architectures with ablation as the causal ground truth, reveals a finding not accessible to prior methods: the architecture families studied organise MLP computation through structurally distinct strategies. In BERT (Devlin et al., 2019) (encoder-only), blocks differentiate by the dimensionality of their transformations, with effective rank increasing across layers while transformation magnitude remains constant. In mBART (Liu et al., 2020; Tang et al., 2020) (encoder-decoder), blocks differentiate by intensity, with transformation magnitude increasing while dimensionality remains constant. In GPT-2 (Radford et al., 2019) (decoder-only), blocks also differentiate by intensity, but late-layer transformations become detrimental. This structural distinction explains the previously unexplained failure of scalar importance measures, such as the L2 norm and the gradient, to generalise across architectures.

The contributions of this article are as follows:

1. A general exact decomposition framework for transformer MLP blocks, which includes GELU and SiLU activation functions, as well as both standard two-matrix and gated three-matrix MLP architectures. This framework preserves the exactness of Re3 while extending its applicability to modern transformers.

2. Verification of exactness across five models spanning two domains (natural language processing and computer vision), using two activation functions (GELU and SiLU), and two MLP structures (standard and gated). The models tested includes BERT-base, mBART-50, GPT-2 Small, ViT-base (Dosovitskiy et al., 2021), and TinyLlama-1.1B (Zhang et al., 2024).

3. A spectral analysis framework for the effective weight matrix $\mathbf{A}_x$ that characterises each MLP block by examining the dimensionality, intensity, and concentration of its transformation. This provides structural information that is inaccessible through gradient-based, probing-based, or projection-based methods.

4. Cross-architecture structural findings demonstrating that encoder-only, encoder-decoder, and decoder-only models utilise different strategies for organising MLP computation. This distinction can only be observed by analysing the exact effective weight matrix.

The remainder of this paper is organised as follows. Section 2 reviews related work on mechanistic interpretability, MLP analysis, and piecewise-affine decomposition. Section 3 presents the soft-gate decomposition for standard and gated MLPs and introduces the spectral analysis framework for $\mathbf{A}_x$. Section 4 reports results across five models, covering verification, importance analysis, and structural characterisation. Section 5 interprets the findings, evaluates the contributions, and discusses limitations. Section 6 concludes the paper and outlines directions for future work.

2 Related Work

Mechanistic interpretability: The circuits framework (Olah et al., 2020; Elhage et al., 2021) aims to reverse-engineer the computations performed by neural networks (NNs) by identifying interpretable circuits made up of individual neurons and attention heads. This line of research has produced detailed explanations of specific mechanisms, such as induction heads and copy-suppression heads in autoregressive models. In contrast, the current work focuses on a different level of analysis: instead of tracing individual circuits, it

characterises the overall structural properties of each MLP block through the effective weight matrix $\mathbf{A}_x$. This approach provides a block-level perspective that complements the neuron-level analysis offered by the circuits framework.

Understanding transformer MLP blocks: Geva et al. (2021) demonstrated that transformer feed-forward layers function as key-value memories. Specifically, individual neurons in the intermediate layers store associations between input patterns (keys) and their corresponding output updates (values). Subsequent work by Geva et al. (2022) showed that MLP blocks promote specific concepts in the vocabulary space, with later blocks refining earlier predictions. These studies characterise what information MLP blocks store and retrieve. The present work addresses a complementary question: how MLP blocks transform their inputs. The effective weight matrix $\mathbf{A}_x$ provides the exact linear mapping from input to output for each block, enabling structural characterisation that includes dimensionality, spectral concentration, and transformation intensity that representation-based analyses do not provide.

Layer-wise interpretability: Several methods analyse transformers by examining how representations evolve across different layers. DecoderLens (Langedijk et al., 2024) projects intermediate decoder representations through successive layers to track how predictions are formed in encoder-decoder models. Probing methods involve training linear classifiers on hidden representations to assess the linguistic or semantic information encoded at each layer (Belinkov et al., 2017). While these approaches focus on the representations at each layer, they do not reveal the transformations applied between them. The soft-gate decomposition addresses this gap by extracting the exact transformation matrix at each layer, allowing for an analysis of how the transformation structure varies across blocks, independent of the representations themselves.

Attribution and importance estimation: Gradient-based methods such as integrated gradients (Sundararajan et al., 2017) attribute model predictions to input features or intermediate components by computing partial derivatives. Ablation studies measure causal importance by removing individual components and observing the effect on model output. Activation patching (Vig et al., 2020) replaces activations at specific positions with those from a different input to trace information flow. These methods quantify the importance of individual components but do not reveal the internal structure of the transformations those components perform. The present study uses ablation as causal ground truth and demonstrates that both gradient-based and output-based importance measures fail to generalise across architectures, a failure that the structural analysis of $\mathbf{A}_x$ explains through the distinction between transformation dimensionality and transformation intensity.

Piecewise-affine decomposition: The piecewise-affine of ReLU networks has a substantial theoretical foundation. ReLU networks divide their input space into linear regions within which the network behaves as an affine map, and the number and structure of these regions have been studied as a measure of network expressivity (Montúfar et al., 2014; Serra et al., 2018). Re3 (Barua et al., 2026) built on this perspective by introducing an exact piecewise-affine decomposition for interpretability, exploiting the binary nature of ReLU activation regions: each input maps to a specific activation mask that determines a linear region, within which the network output is an affine function of the input. The present work generalises this framework from binary gates ($\mathbf{1}_{[z>0]}$) to continuous soft gates ($\Phi(z)$ for GELU, $\sigma(z)$ for SiLU), preserving the exact piecewise-affine structure while extending applicability to modern transformer architectures. The further extension to gated MLP with separate gate and value projections (Proposition 2) broadens the framework to include the gated three-matrix MLP structure used in architectures of the LLaMA-family (Touvron et al., 2023).

Distinction from prior work: The methods mentioned above analyse transformer components using various approaches, including representations (such as probing and layer-wise projection), sensitivity (including gradients and activation patching), removal effects (through ablation), and neuron-level circuit tracing. Singular value decomposition (SVD) has also been applied to the static, input-independent weight matrices of transformer components to identify interpretable directions. However, none of these methods provides the precise linear transformation that each MLP block applies to each input. The soft-gate decomposition addresses this issue by extracting the effective weight matrix $\mathbf{A}_x \in \mathbb{R}^{d\times d}$, which encodes the complete

input-to-output mapping of a block for a specific input. Unlike the spectral analysis of static weights, the decomposition of the input-dependent $\mathbf{A}_x$ captures how the transformation performed by a block varies across different inputs and layers. The SVD-based analysis of $\mathbf{A}_x$ separates structural properties such as transformation dimensionality, intensity, and spectral concentration capabilities that representation-based and sensitivity-based methods do not provide. This structural separation is what enables the cross-architecture findings reported in this paper.

3 Soft-Gate Decomposition Framework for Transformer MLPs

This section presents the soft-gate decomposition framework for transformer MLP blocks. Section 3.1 derives the decomposition for two-matrix MLPs with GELU activation, while Section 3.2 extends it to gated three-matrix MLPs using SiLU activation. Section 3.3 offers a spectral analysis of the effective weight matrix $\mathbf{A}_x$. The symbol $\odot$ denotes the Hadamard (element-wise) product.

3.1 Soft-Gate Decomposition for Standard MLPs

Within each transformer layer, the MLP block receives its input from the residual stream, following the attention sub-layer and layer normalisation. Consider a standard transformer MLP block parameterised by weight matrices $\mathbf{W}_1 \in \mathbb{R}^{m\times d}$, $\mathbf{W}_2 \in \mathbb{R}^{d\times m}$ and bias vectors $\mathbf{b}_1 \in \mathbb{R}^m$, $\mathbf{b}_2 \in \mathbb{R}^d$, where d denotes the model dimension and m denotes the intermediate (feed-forward) dimension. Given an input $\mathbf{x} \in \mathbb{R}^d$, the MLP computes:

$$\mathrm{MLP}(\mathbf{x}) = \mathbf{W}_2\, \phi(\mathbf{W}_1\mathbf{x} + \mathbf{b}_1) + \mathbf{b}_2, \tag{1}$$

where ϕ is an element-wise activation function. In BERT, GPT-2, mBART, and ViT, the activation is GELU (Hendrycks & Gimpel, 2016), defined as:

$$\mathrm{GELU}(z) = z \cdot \Phi(z), \tag{2}$$

where $\Phi(\cdot)$ denotes the cumulative distribution function of the standard Gaussian distribution. The key observation underlying the decomposition is that GELU admits a multiplicative factorisation into a linear term z and a *soft gate* $\Phi(z) \in (0, 1)$.

Definition 1 (Soft Gate)**.** *For a pre-activation vector $\mathbf{z} = \mathbf{W}_1\mathbf{x} + \mathbf{b}_1 \in \mathbb{R}^m$, the soft gate vector is defined as $\mathbf{g} = \Phi(\mathbf{z}) \in (0, 1)^m$, where Φ is applied element-wise. Each component $g_i = \Phi(z_i)$ represents the degree of activation of the i-th intermediate neuron, varying continuously between fully inactive ($g_i \approx 0$) and fully active ($g_i \approx 1$).*

Substituting equation 2 into equation 1 and introducing the soft gate yields:

$$\begin{aligned}
\mathrm{MLP}(\mathbf{x}) &= \mathbf{W}_2\big(\mathbf{z} \odot \mathbf{g}\big) + \mathbf{b}_2 \\
&= \mathbf{W}_2\, \mathrm{diag}(\mathbf{g})\, \mathbf{z} + \mathbf{b}_2 \\
&= \mathbf{W}_2\, \mathrm{diag}(\mathbf{g})\, (\mathbf{W}_1\mathbf{x} + \mathbf{b}_1) + \mathbf{b}_2 \\
&= \underbrace{\mathbf{W}_2\, \mathrm{diag}(\mathbf{g})\, \mathbf{W}_1}_{\mathbf{A}_x}\, \mathbf{x} \;+\; \underbrace{\mathbf{W}_2\, \mathrm{diag}(\mathbf{g})\, \mathbf{b}_1 + \mathbf{b}_2}_{\mathbf{D}_x}.
\end{aligned} \tag{3}$$

Proposition 1 (Exact Decomposition for Standard MLPs)**.** *For any input $\mathbf{x} \in \mathbb{R}^d$, the output of a standard MLP block with GELU activation satisfies:*

$$\mathit{MLP}(\mathbf{x}) = \mathbf{A}_x\, \mathbf{x} + \mathbf{D}_x, \tag{4}$$

where $\mathbf{A}_x = \mathbf{W}_2\, \mathit{diag}(\Phi(\mathbf{W}_1\mathbf{x} + \mathbf{b}_1))\, \mathbf{W}_1 \in \mathbb{R}^{d\times d}$ is the effective weight matrix and $\mathbf{D}_x = \mathbf{W}_2\, \mathit{diag}(\Phi(\mathbf{W}_1\mathbf{x} + \mathbf{b}_1))\, \mathbf{b}_1 + \mathbf{b}_2 \in \mathbb{R}^d$ is the effective bias vector. This decomposition is exact with zero mathematical approximation error.

The matrix $\mathbf{A}_x \in \mathbb{R}^{d\times d}$ encodes the complete input-to-output linear mapping of the MLP block at input $\mathbf{x}$. Element (i, j) of $\mathbf{A}_x$ specifies the exact contribution of input feature j to output feature i. Both $\mathbf{A}_x$ and $\mathbf{D}_x$ depend on $\mathbf{x}$ through the soft gate $\mathbf{g}$, making the decomposition input-dependent. For each distinct input, a different effective weight matrix governs the transformation.

Remark 1 (Generality)**.** *The decomposition in Proposition 1 holds for any activation function that admits a multiplicative factorisation $\phi(z) = z \cdot \gamma(z)$, where $\gamma : \mathbb{R} \to \mathbb{R}$ is an arbitrary function. Setting $\gamma(z) = \Phi(z)$ recovers GELU. Setting $\gamma(z) = \mathbf{1}_{[z>0]}$ recovers ReLU, reducing the framework to Re3 (Barua et al., 2026) as a special case with binary gates.*

In a transformer with L layers, the decomposition is applied independently to each MLP block $l \in \{1, \ldots, L\}$, yielding a layer-specific effective weight matrix $\mathbf{A}_x^{(l)}$. This matrix varies across blocks due to differences in both the learned weights ($\mathbf{W}_1^{(l)}$, $\mathbf{W}_2^{(l)}$) and the input $\mathbf{x}^{(l)}$, which itself depends on all preceding attention and MLP computations through the residual stream.

3.2 Extension to Gated MLPs

Modern decoder-only architectures such as those in the LLaMA family (Touvron et al., 2023) employ a gated MLP structure with three weight matrices and SiLU activation (Elfwing et al., 2018). The gated MLP takes the form:

$$\mathrm{MLP}(\mathbf{x}) = \mathbf{W}_d\big(\mathrm{SiLU}(\mathbf{W}_g\mathbf{x}) \odot (\mathbf{W}_u\mathbf{x})\big), \tag{5}$$

where $\mathbf{W}_g \in \mathbb{R}^{m\times d}$ is the gate projection, $\mathbf{W}_u \in \mathbb{R}^{m\times d}$ is the up projection, $\mathbf{W}_d \in \mathbb{R}^{d\times m}$ is the down projection, and SiLU is defined as $\mathrm{SiLU}(z) = z \cdot \sigma(z)$ with σ denoting the logistic sigmoid (Elfwing et al., 2018; Ramachandran et al., 2017). These architectures typically omit bias terms.

The gate and value projections in equation 5 are computed from the same input $\mathbf{x}$ but through different weight matrices. Defining the effective gate vector $\mathbf{h} = \mathrm{SiLU}(\mathbf{W}_g\mathbf{x}) \in \mathbb{R}^m$, the output can be rewritten as:

$$\begin{aligned}\mathrm{MLP}(\mathbf{x}) &= \mathbf{W}_d\big(\mathbf{h} \odot (\mathbf{W}_u\mathbf{x})\big)\\ &= \mathbf{W}_d\,\mathrm{diag}(\mathbf{h})\,\mathbf{W}_u\,\mathbf{x}\\ &= \underbrace{\mathbf{W}_d\,\mathrm{diag}(\mathbf{h})\,\mathbf{W}_u}_{\mathbf{A}_x}\,\mathbf{x}.\end{aligned} \tag{6}$$

Proposition 2 (Exact Decomposition for Gated MLPs)**.** *For any input $\mathbf{x} \in \mathbb{R}^d$, the output of a gated MLP block with SiLU activation satisfies:*

$$MLP(\mathbf{x}) = \mathbf{A}_x\,\mathbf{x}, \tag{7}$$

where $\mathbf{A}_x = \mathbf{W}_d\,diag(SiLU(\mathbf{W}_g\mathbf{x}))\,\mathbf{W}_u \in \mathbb{R}^{d\times d}$. The effective bias $\mathbf{D}_x = \mathbf{0}$ when the MLP contains no bias parameters.

The structure of the decomposition is identical in both cases: a $d \times d$ effective weight matrix $\mathbf{A}_x$ constructed by gating one projection matrix with input-dependent coefficients and multiplying by another. The standard and gated variants differ in two respects. First, the gate function changes from $\Phi(z) \in (0, 1)$ to $\mathrm{SiLU}(z) = z\cdot\sigma(z) \in \mathbb{R}$, which is unbounded and can take negative values (minimum approximately -0.278 at $z \approx -1.28$). The activation semantics of Definition 1, which interpret each gate value as a degree of activation in $(0, 1)$, therefore apply only to the bounded GELU case; for the gated SiLU case, $\mathbf{h}$ acts as a multiplicative coefficient that may be negative. Second, the gate and value projections originate from different weight matrices ($\mathbf{W}_g$ and $\mathbf{W}_u$) rather than a single shared matrix ($\mathbf{W}_1$).

3.3 Spectral Analysis of $\mathbf{A}_x$

The effective weight matrix $\mathbf{A}_x \in \mathbb{R}^{d\times d}$ encodes the complete linear transformation applied by an MLP block at a given input. To characterise the structural properties of this transformation, an SVD of $\mathbf{A}_x$ is performed:

$$\mathbf{A}_x = \mathbf{U}\,\mathbf{\Sigma}\,\mathbf{V}^\top, \tag{8}$$

where $\mathbf{\Sigma} = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_d)$ contains the singular values in non-increasing order $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_d \geq 0$. The columns of $\mathbf{V}$ represent the input directions being read from the residual stream, and the columns of $\mathbf{U}$ represent the output directions being written to the residual stream. The singular values σ_i quantify the amplification along each direction.

Algorithm 1 Soft-Gate Decomposition for Standard MLP (GELU)

Require: Weight matrices $\mathbf{W}_1 \in \mathbb{R}^{m\times d}$, $\mathbf{W}_2 \in \mathbb{R}^{d\times m}$; biases $\mathbf{b}_1 \in \mathbb{R}^m$, $\mathbf{b}_2 \in \mathbb{R}^d$; input $\mathbf{x} \in \mathbb{R}^d$
Ensure: Effective weight matrix $\mathbf{A}_x$, effective bias $\mathbf{D}_x$, MLP output, gate vector $\mathbf{g}$

1: $\mathbf{z} \leftarrow \mathbf{W}_1\mathbf{x} + \mathbf{b}_1$ ▷ Pre-activation, $\mathbf{z} \in \mathbb{R}^m$
2: $\mathbf{g} \leftarrow \Phi(\mathbf{z})$ ▷ Soft gate (element-wise Gaussian CDF), $\mathbf{g} \in (0,1)^m$
3: $\widetilde{\mathbf{W}}_2 \leftarrow \mathbf{W}_2\,\mathrm{diag}(\mathbf{g})$ ▷ Column-scale $\mathbf{W}_2$ by gate values
4: $\mathbf{A}_x \leftarrow \widetilde{\mathbf{W}}_2\,\mathbf{W}_1$ ▷ Effective weight matrix, $\mathbf{A}_x \in \mathbb{R}^{d\times d}$
5: $\mathbf{D}_x \leftarrow \widetilde{\mathbf{W}}_2\,\mathbf{b}_1 + \mathbf{b}_2$ ▷ Effective bias, $\mathbf{D}_x \in \mathbb{R}^d$
6: **return** $\mathbf{A}_x$, $\mathbf{D}_x$, $\mathbf{A}_x\mathbf{x} + \mathbf{D}_x$, $\mathbf{g}$

Six scalar metrics are derived from the SVD to summarise the structural properties of $\mathbf{A}_x$. **Frobenius norm:** the total magnitude of the transformation, computed as $\|\mathbf{A}_x\|_F = \sqrt{\sum_{i=1}^d \sigma_i^2}$. This captures the overall energy of the mapping regardless of its directional distribution. **Spectral norm:** the maximum amplification factor, given by the largest singular value σ_1. A high spectral norm indicates that the transformation concentrates energy along a single dominant direction. **Nuclear norm:** the sum of all singular values $\sum_{i=1}^d \sigma_i$. Unlike the Frobenius norm, which squares the singular values and thus amplifies dominant components, the nuclear norm weights all directions linearly and is sensitive to the number of active directions. **Effective rank:** following Roy & Vetterli (2007), the effective rank quantifies the number of significant dimensions in the transformation,

$$\mathrm{erank}(\mathbf{A}_x) = \exp\Big(-\sum_{i=1}^{d} p_i \log p_i\Big), \qquad p_i = \frac{\sigma_i}{\sum_{j=1}^d \sigma_j}, \tag{9}$$

where the normalised singular values p_i form a probability distribution and the exponentiated Shannon entropy yields a continuous measure of dimensionality. An effective rank of k indicates that the transformation operates in approximately k independent directions. **Stable rank:** a ratio-based measure of dimensionality defined as $\|\mathbf{A}_x\|_F^2/\sigma_1^2$, representing the number of copies of the dominant singular direction required to account for the total energy. **Trace:** the sum of diagonal entries $\mathrm{tr}(\mathbf{A}_x) = \sum_{i=1}^d [\mathbf{A}_x]_{ii}$, measuring the degree to which the transformation maps each feature to itself. A large positive trace indicates feature preservation; a trace near zero indicates feature mixing. These six metrics provide complementary perspectives on the same transformation. The Frobenius norm and nuclear norm measure overall magnitude. The spectral norm and stable rank characterise concentration. The effective rank measures dimensionality. The trace captures self-mapping structure.

In practice, each metric is computed per block and per input. To obtain a stable characterisation of each block, the metrics are averaged across a corpus of N inputs. The resulting per-block averages serve as structural descriptors of how each MLP block organises its computation, independent of any single input.

3.4 Algorithms

Algorithms 1 and 2 describe the decomposition procedures for standard and gated MLPs, respectively. Both algorithms construct the effective weight matrix $\mathbf{A}_x$ and return the exact MLP output and the gate vector for downstream analysis. Constructing $\mathbf{A}_x$ requires a matrix-matrix multiplication costing $\mathcal{O}(d^2m)$, approximately d times more expensive than the two matrix-vector multiplications in a standard forward pass. The full construction is therefore invoked only when the structural properties of $\mathbf{A}_x$ are required, as in the spectral analysis of Section 3.3. Analyses requiring only scalar summaries, such as the L2 norm of the output, use a fast path that bypasses $\mathbf{A}_x$.

4 Experiments and Results

The decomposition framework is evaluated across five transformer models spanning two domains, two activation functions, and two MLP architectures. The interpretability analysis focuses on three models for

Algorithm 2 Soft-Gate Decomposition for Gated MLP (SiLU)

Require: Gate projection $\mathbf{W}_g \in \mathbb{R}^{m\times d}$; up projection $\mathbf{W}_u \in \mathbb{R}^{m\times d}$; down projection $\mathbf{W}_d \in \mathbb{R}^{d\times m}$; input $\mathbf{x} \in \mathbb{R}^d$

Ensure: Effective weight matrix $\mathbf{A}_x$, MLP output, gate vector $\mathbf{h}$

1: $\mathbf{h} \leftarrow \mathrm{SiLU}(\mathbf{W}_g\mathbf{x})$ ▷ Effective gate (element-wise), $\mathbf{h} \in \mathbb{R}^m$

2: $\widetilde{\mathbf{W}}_d \leftarrow \mathbf{W}_d \,\mathrm{diag}(\mathbf{h})$ ▷ Column-scale $\mathbf{W}_d$ by gate values

3: $\mathbf{A}_x \leftarrow \widetilde{\mathbf{W}}_d\,\mathbf{W}_u$ ▷ Effective weight matrix, $\mathbf{A}_x \in \mathbb{R}^{d\times d}$

4: **return** $\mathbf{A}_x$, $\mathbf{A}_x\mathbf{x}$, $\mathbf{h}$

Table 1: Models used for evaluation. The top three are analysed using the full pipeline, including ablation, and the bottom two verify the generality of the decomposition across domains and across MLP architectures.

Model	Architecture	Activation	MLP Type	d	m	Layers	Task
BERT-base	Encoder	GELU	Standard	768	3072	12	Masked LM
mBART-50	Enc-Dec[1]	GELU	Standard	1024	4096	24	Translation
GPT-2 Small	Decoder	GELU	Standard	768	3072	12	Next-token
ViT-base	Encoder	GELU	Standard	768	3072	12	Classification
TinyLlama-1.1B	Decoder	SiLU	Gated	2048	5632	22	Next-token

which full causal ground truth (ablation) is available: BERT-base (Devlin et al., 2019), mBART-50 (Tang et al., 2020), and GPT-2 Small (Radford et al., 2019). Two additional models, ViT-base (Dosovitskiy et al., 2021) and TinyLlama-1.1B (Zhang et al., 2024), serve to verify the generality of the decomposition across vision transformers and gated MLP architectures, respectively.

4.1 Experimental Setup

Table 1 summarises the five models under study. The three interpretability models represent the main transformer families: encoder-only (BERT), encoder-decoder (mBART), and decoder-only (GPT-2). Each model is evaluated based on its specific task using established benchmarks: BERT for masked language modelling on WikiText-103 (Merity et al., 2017), mBART for English-to-German translation on WMT14 (Bojar et al., 2014), and GPT-2 for next-token prediction on WikiText-103. ViT is assessed on the CIFAR-10 image classification task (Krizhevsky, 2009), while TinyLlama is evaluated on next-token prediction with WikiText-103.

For each interpretability model, 200 samples are processed through the decomposition pipeline. MLP input at each position is captured using forward hooks, and the decomposition method (as described in Section 3) is applied to all MLP blocks. In BERT, analysis focuses on the `[MASK]` token; for mBART, encoder representations are averaged across source positions, while decoder representations are averaged for each generated token. For GPT-2, all token positions are analysed and averaged per sentence. An ablation approach assesses the importance of each causal block by setting the output of each MLP block to zero and measuring the resulting change in the target logits. A positive logit drop indicates a contribution to correct predictions, while a negative drop indicates a detrimental effect. This importance measure serves as ground truth for validating decomposition-based measures using Spearman's rank correlation. The fast-path decomposition processes samples 1.5 to 4 times faster than full ablation, depending on the model. Full experimental details are in Appendix A.

4.2 Verification of Exactness

The decomposition is verified by comparing the reconstructed output $\mathbf{A}_x\mathbf{x} + \mathbf{D}_x$ against the actual MLP output computed through the standard forward pass, for every block and every sample. A block-sample pair

[1]mBART-50 comprises 12 encoder and 12 decoder layers. The interpretability analysis uses the 12 decoder blocks.

Table 2: Verification of decomposition exactness. Max and mean errors report the worst-case block per model, averaged across samples. Pass rate is computed across all blocks and samples per model.

Model	Blocks	Samples	Max Error	Mean Error	Pass Rate
BERT-base	12	200	8.86×10^{-7}	5.25×10^{-8}	100%
mBART-50	24	200	7.04×10^{-6}	1.12×10^{-7}	100%
GPT-2 Small	12	200	8.29×10^{-6}	7.38×10^{-7}	100%
ViT-base	12	200	1.49×10^{-6}	2.64×10^{-7}	100%
TinyLlama-1.1B	22	200	8.25×10^{-6}	1.41×10^{-7}	99%*

*TinyLlama layer 21 shows isolated cases (2 of 200 samples) in which the maximum error reaches 1.37×10^{-4}, marginally exceeding the threshold due to float32 precision accumulation across 22 layers. The mean error on this layer remains at 1.41×10^{-7}.

Table 3: Spearman rank correlation (ρ) between importance measures and ablation ground truth. BERT and GPT-2 use all 12 blocks; mBART uses the 11 decoder blocks, excluding decoder 11 (a pre-norm outlier). Bold indicates $|\rho| > 0.75$ with $p < 0.05$.

Measure	BERT	mBART (dec.)	GPT-2
L2 Norm	**−0.80**	**+0.79**	**−0.88**
|DLA|	**−0.85**	+0.42	**−0.87**
Gradient	**+0.95**	−0.59	**+0.86**
Gate activity	**+0.95**	−0.75	−0.46
Random	≈ 0.00	≈ 0.00	≈ 0.00

is classified as passing if the maximum absolute element-wise error falls below 10^{-4}. Table 2 summarises the results across all five models.

Across 82 MLP blocks and 1,000 verification runs, the reconstruction error remains at the level of floating-point arithmetic precision. For the four standard-MLP models, all errors fall within the range 10^{-8} to 10^{-6}. The gated MLP decomposition (Proposition 2) achieves errors in the same range on TinyLlama, except at layer 21, where 2 of 200 samples produce a maximum error of 1.37×10^{-4}, while the mean error remains at 1.41×10^{-7}. Layer 7 of TinyLlama shows a larger standard deviation in its maximum error (1.43×10^{-5}) arising from isolated high-magnitude inputs, though all 200 samples for this layer remain below the 10^{-4} threshold. Per-block error distributions for all models are provided in Appendix B.

4.3 Block Importance Analysis

Causal block importance is determined through ablation, as outlined in Section 4.1, resulting in three distinct importance profiles across the architectures. In BERT, the importance increases monotonically with depth, with layer 10 producing the largest logit drop (+1.01). In the mBART decoder, importance is concentrated in layers 8 to 10, while decoder layer 11 shows a negative logit drop of -0.23. In GPT-2, the middle layers generate the largest positive logit drops (layer 3: +2.47), whereas the late layers exhibit significant negative drops (layer 10: -16.4, layer 11: -10.5). Full per-block ablation values can be found in Appendix D. Four importance measures are evaluated against this ground truth: the L2 norm and direct logit attribution of the decomposed MLP output, gate activity (the percentage of partially active neurons in the soft gate vector $\mathbf{g}$), and gradient-based attribution as an external baseline.

Table 3 presents the Spearman rank correlations. The L2 norm shows a positive correlation in the mBART decoder ($\rho = +0.79$), but a negative correlation in both BERT ($\rho = -0.80$) and GPT-2 ($\rho = -0.88$). Gradient-based attribution has a positive correlation with BERT ($\rho = +0.95$) and GPT-2 ($\rho = +0.86$), but a negative correlation with mBART ($\rho = -0.59$). Gate activity reaches $\rho = +0.95$ on BERT but shows $\rho = -0.46$ (not significant) on GPT-2. No single measure achieves consistent positive correlation across all three architectures.

Table 4: Spearman rank correlation (ρ) between $\mathbf{A}_x$-derived structural metrics and ablation ground truth. mBART reports decoder blocks excluding decoder_11. Bold indicates $|\rho| > 0.75$ with $p < 0.05$.

Metric	BERT	mBART (dec.)	GPT-2
$\|\mathbf{A}_x\|_F$	$+0.55$	$\mathbf{+0.95}$	$\mathbf{-0.81}$
$\sigma_{\max}$	$\mathbf{-0.94}$	$+0.72$	-0.55
Nuclear norm	$\mathbf{+0.97}$	$\mathbf{+0.96}$	$\mathbf{-0.77}$
Effective rank	$\mathbf{+0.98}$	-0.32	$+0.30$
Stable rank	$\mathbf{+0.95}$	-0.61	$+0.15$
Trace	-0.31	$\mathbf{-0.93}$	$+0.01$

4.4 Structural Analysis of the Effective Weight Matrix

The preceding analysis measured properties of the MLP output vector. This section examines the effective weight matrix $\mathbf{A}_x$ itself, applying the SVD-based spectral analysis defined in Section 3.3. For each of the three interpretability models, $\mathbf{A}_x$ is constructed for every MLP block, and 200 samples are drawn per model. Six structural metrics are extracted per block. Across all blocks and models, the coefficient of variation of the effective rank remains below 6%. Full per-block metrics and variability tables are provided in Appendix H.

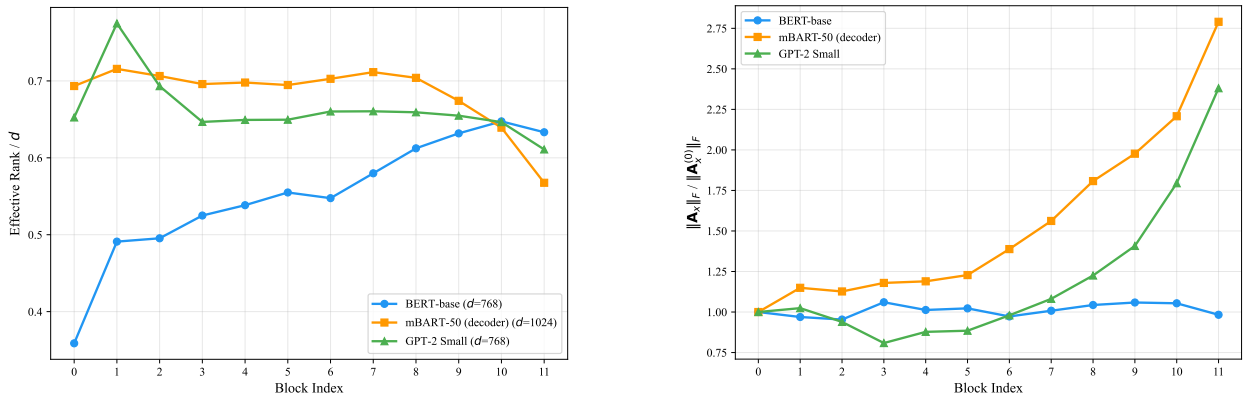
Correlation with ablation: Table 4 reports Spearman correlations between each $\mathbf{A}_x$-derived metric and the ablation ground truth. On BERT, the effective rank achieves $\rho = +0.98$ ($p < 0.05$). The nuclear norm ($\rho = +0.97$) and stable rank ($\rho = +0.95$) are also significant. However, the Frobenius norm is not significant ($\rho = +0.55$). On the mBART decoder, the nuclear norm ($\rho = +0.96$) and Frobenius norm ($\rho = +0.95$) show the strongest correlations, while the effective rank shows no significant relationship ($\rho = -0.32$). On GPT-2, the Frobenius norm ($\rho = -0.81$) and nuclear norm ($\rho = -0.77$) are both significant and negative. The effective rank shows no significant correlation on GPT-2 ($\rho = +0.30$).

Spectral Profiles: Figure 1 illustrates the structural profiles of $\mathbf{A}_x$ across layers for all three models. In Figure 1(a), the effective rank is normalised by the model dimension d. In BERT, the normalised effective rank increases from 0.36 at layer 0 (indicating 276 out of 768 dimensions) to 0.65 at layer 10 (497 dimensions). The top singular value accounts for 5.9% of the nuclear norm at layer 0 and 1.1% at layer 10. In the mBART decoder, the normalised effective rank consistently ranges from 0.67 to 0.72 across layers 0 to 9, with a slight decrease to 0.64 at decoder layer 10 and 0.57 at decoder layer 11. In GPT-2, the normalised effective rank remains between 0.64 and 0.66 across layers 3 to 10, with layer 1 as an outlier at 0.78 and layer 11 dropping to 0.61. Figure 1(b) displays the Frobenius norm relative to the first block. In BERT, this ratio stays between 0.95 and 1.06 across all layers. In the mBART decoder, the value shows a monotonic increase from 1.0 to 2.21 at decoder layer 10, then further increases to 2.79 at decoder layer 11. In GPT-2, the ratio decreases slightly to 0.81 at layer 3, then rises to 2.38 at layer 11.

Summary of observations: The results show two contrasting patterns across architectures. In BERT, the effective rank varies substantially across blocks (276–497) while the Frobenius norm remains approximately constant (≈ 12). In both mBART and GPT-2, the effective rank is approximately constant (≈ 700 and ≈ 500 respectively) while the Frobenius norm increases substantially with depth. On BERT, the dimensionality-related metrics (effective rank, stable rank) correlate positively with ablation importance. On mBART, the magnitude-related metrics (Frobenius norm, nuclear norm) correlate positively with ablation importance. On GPT-2, magnitude-related metrics are significantly correlated with ablation but with a negative sign, and the blocks with the highest Frobenius norm have negative ablation values.

5 Discussion

The primary goal of this work is to provide a detailed mechanistic decomposition of transformer MLP blocks that offers interpretability beyond what is available with existing methods. This decomposition is achieved



(a) Normalised effective rank

(b) Relative Frobenius norm

Figure 1: Structural profiles of the effective weight matrix across layers where (a) is the effective rank normalised by model dimension and (b) is the Frobenius norm relative to the first block.

by extracting the effective weight matrix $\mathbf{A}_x \in \mathbb{R}^{d \times d}$, which encodes the complete linear mapping from input to output for each MLP block at every input. The results presented in Section 4 demonstrate three key outcomes: first, the decomposition is exact across all tested architectures (as shown in Table 2); second, the derived measures effectively characterize block-level behavior in a way that correlates with causal importance (as indicated in Tables 3 and 4); and third, the spectral analysis of $\mathbf{A}_x$ reveals structural properties of MLP computation that vary across different architecture families (illustrated in Figure 1).

Generality and reliability of the framework: The decomposition improves upon Re3 (Barua et al., 2026) by extending its usage from ReLU to any activation function of the form $\phi(z) = z \cdot \gamma(z)$. This is achieved by replacing the binary gate with a continuous gate $\gamma(z)$. The extension to gated MLPs, as outlined in Proposition 2, allows for separate weight matrices to be used for gate and value projections, thereby broadening the framework to include the LLaMA family (Touvron et al., 2023). Verification across five models shows that reconstruction errors fall within float32 arithmetic precision (ranging from 10^{-8} to 10^{-6}) for 82 MLP blocks over 1,000 verification runs (see Table 2). The method is fully deterministic, yielding identical outputs for the same inputs (with Spearman correlation $\rho = 1.000$ across 50 repeated runs on each model), unlike gradient-based methods that require backward-pass computation. Block importance rankings are consistent across various samples; the highest-ranked block maintains its position in over 90% of samples across all three interpretability models. Fast-path computation, which bypasses the construction of $\mathbf{A}_x$, processes each sample 1.5 to 4 times faster than full ablation, making the framework practical for routine analyses. Collectively, these features establish the decomposition as a reliable, reproducible, and efficient tool for MLP-level interpretability.

Interpretation of gate and output-level findings: The gate vector $\mathbf{g} = \Phi(\mathbf{z})$ provides the first interpretive layer of the decomposition. The three models exhibit qualitatively distinct gating behaviour: BERT shows gate activity increasing from 20.8% to 65.5% of partially active neurons with depth, suggesting progressively broader neuron engagement as representations are refined towards the masked token prediction. The mBART decoder shows the opposite overall trend, with gate activity rising to a peak near decoder 3 (95.1%) and then decreasing to 70.9% at decoder 10, consistent with increasing selectivity in the later decoder layers as the model narrows towards a specific target token. GPT-2 maintains approximately flat gate activity (68% to 83%), indicating that the gating mechanism does not strongly differentiate blocks in this architecture. These differences in gating strategy arise despite all three models using the same activation function (GELU), pointing to the influence of training objective and architectural role (encoder vs decoder) on the learned gating behaviour.

At the output level, the L2 norm and direct logit attribution of the MLP output, together with gradient-based attribution, show architecture-dependent relationships with causal importance (Table 3). The L2 norm correlates positively with ablation importance on the mBART decoder ($\rho = +0.79$) but negatively on BERT ($\rho = -0.80$) and GPT-2 ($\rho = -0.88$). Gradient-based attribution achieves $\rho = +0.95$ on BERT and $\rho = +0.86$ on GPT-2, but it inverts on mBART ($\rho = -0.59$). The gradient inversion on mBART is consistent with the observation that the most important decoder blocks (layers 8–10) apply strong transformations with high Frobenius norm: gradient measures sensitivity to perturbation, which need not align with causal contribution when blocks of varying intensity are involved. A block applying a strong transformation may tolerate perturbation better than a block applying a weak one, inverting the sensitivity ranking relative to the importance ranking.

The interpretive capability of $\mathbf{A}_x$: The effective weight matrix $\mathbf{A}_x$ provides a second, deeper interpretive layer unique to the exact decomposition. Gradient-based methods quantify the sensitivity of the output to perturbations of each block but do not reveal the internal structure of the transformation: how many independent directions the block uses, whether its energy is concentrated or distributed, or whether it preserves or mixes input features. Probing methods (Geva et al., 2021; 2022) measure representational content at each layer by training auxiliary classifiers, but do not examine the transformation between representations. The six SVD-derived metrics (Section 3.3) characterise complementary aspects of the transformation structure from a single object, $\mathbf{A}_x$. The coefficient of variation of the effective rank remains below 6% across all blocks and all three models, and that of the Frobenius norm remains below 14%, indicating that these spectral properties are stable characteristics of each block that persist across diverse inputs.

Cross-architecture structural organisation: The spectral analysis of $\mathbf{A}_x$ shows that the three architecture families studied organise MLP computation through structurally distinct strategies. In BERT (encoder-only), blocks differ by *dimensionality*: the effective rank increases from 276 at layer 0 to 497 at layer 10, while the Frobenius norm remains approximately constant (≈ 12). Early blocks concentrate transformation energy along a few directions (the top singular value captures 5.9% of the nuclear norm at layer 0), whereas late blocks distribute energy more broadly (1.1% at layer 10). The correlation between effective rank and ablation importance ($\rho = +0.98$) indicates that blocks operating in higher-dimensional subspaces are those whose removal most disrupts model predictions. This dimensional progression aligns with a processing hierarchy in which early layers perform coarse, focused transformations and later layers perform fine-grained, distributed refinement.

In the mBART decoder (encoder-decoder), blocks differentiate by *intensity*: the Frobenius norm increases from 42.9 at decoder 0 to 94.8 at decoder 10 while the effective rank remains approximately constant (≈ 700). All blocks operate in a comparable number of dimensions, but late blocks apply substantially stronger transformations. The correlation between the nuclear norm and ablation importance ($\rho = +0.96$) indicates that transformation magnitude tracks causal contribution. This intensity progression is consistent with the demands of cross-lingual generation, where later decoder layers must apply increasingly forceful transformations to map source-language representations into target-language token predictions.

In GPT-2 (decoder-only), blocks share the intensity differentiation pattern of mBART: the Frobenius norm increases from 127 to 375 while the effective rank remains approximately constant (≈ 500). However, the relationship with ablation importance is inverted. The late blocks with the highest Frobenius norms (layer 10: 283, layer 11: 375) correspond to the most negative ablation values (-16.4 and -10.5), indicating that their presence is detrimental. These blocks also exhibit the highest spectral concentration ($\sigma_{\max} = 129$ and 215), meaning their transformations are dominated by a single direction. The causally important middle blocks (layers 1 and 3) apply moderate transformations without such concentration. This pattern is consistent with over-specialisation in the autoregressive generation process: late blocks develop intense, narrowly focused transformations that sharpen the output distribution beyond the point of benefit, and their removal allows the more balanced middle-layer representations to pass through.

Explaining the failure of scalar importance measures: The architecture-dependent failure of scalar measures (Table 3) follows directly from the structural differences revealed by the spectral analysis. On BERT, the L2 norm of the MLP output is largest in early layers, where $\sigma_{\max}$ is high (7.95 at layer 0) and

the effective rank is low (276). These blocks amplify the input along a small number of directions, producing a large output without operating across many dimensions. Removing such a block has a limited causal impact because its narrow transformation is easier for other blocks to compensate for. On mBART, the L2 norm tracks importance because intensity genuinely reflects causal contribution in that architecture. On GPT-2, the L2 norm is inverted because the most intense blocks are harmful. The key distinction between these cases is whether the output magnitude arises from broad, distributed computation (many directions, each moderate) or from narrow, concentrated computation (few directions, each strong). The L2 norm of the output cannot distinguish these cases. The effective weight matrix $\mathbf{A}_x$ can be decomposed into the Frobenius norm (total magnitude), the effective rank (number of active directions), and the spectral norm (concentration in the dominant direction).

Relation to prior work: The circuits framework (Elhage et al., 2021; Olah et al., 2020) identifies specific computational circuits by analysing individual neurons and attention heads. The $\mathbf{A}_x$ analysis operates at a complementary level of granularity, characterising the aggregate structural properties of each block (dimensionality, intensity, spectral concentration) rather than tracing individual circuits. Geva et al. (2021) characterised feed-forward layers as key-value memories in which individual neurons store and retrieve factual associations. The spectral analysis extends this perspective by revealing that the collective organisation of these memories differs across architectures: BERT arranges them in progressively higher-dimensional subspaces, mBART applies them with progressively greater intensity, and GPT-2 develops excessive intensity in late layers. DecoderLens (Langedijk et al., 2024) tracks prediction formation by projecting intermediate decoder representations through subsequent layers. The decomposition provides a complementary view: rather than projecting representations forward, it extracts the exact transformation applied at each layer, enabling structural characterisation that projection-based methods do not offer.

Limitations: The current study presents several limitations. The interpretability analysis includes models with up to 611 million parameters (mBART-50), while the verification extends to models with 1.1 billion parameters (TinyLlama). The computational cost of constructing the matrix $\mathbf{A}_x$ scales as $\mathcal{O}(d^2m)$ per block, which becomes significant for models with hidden dimensions greater than 4096. Only MLP blocks are decomposed; the attention mechanism, another major component of each transformer layer, is not included in this framework. Each architecture family is represented by a single model, and it remains to be seen whether the observed structural strategies generalise across families, such as from GPT-2 Small to larger decoder-only models. Additionally, no downstream applications are demonstrated; the structural findings are descriptive rather than prescriptive, and it remains unclear whether spectral profiles can inform practical decisions, such as layer pruning or knowledge localisation. Lastly, no universal measure of structural importance has been identified: the relationship between the properties of $\mathbf{A}_x$ and causal importance varies by architecture. While dimensionality-based metrics are significant for BERT and magnitude-based metrics for mBART, neither consistently captures importance in GPT-2.

6 Conclusion

Transformer MLP blocks apply nonlinear transformations whose internal structure remains inaccessible to gradient-based, probing, and attention-based interpretability methods. This paper introduced an exact soft-gate decomposition that extracts the effective weight matrix $\mathbf{A}_x$ for each MLP block at each input, extending the Re3 framework to GELU and SiLU activations and to gated MLP architectures, with verification across five models in two domains. The spectral analysis of $\mathbf{A}_x$ revealed that the encoder-only, encoder-decoder, and decoder-only models studied organise MLP computation through structurally distinct strategies, distinguished by whether blocks vary in transformation dimensionality or transformation intensity. This cross-architecture distinction, visible only through the exact effective weight matrix, provides a structural explanation for the previously unexplained failure of scalar importance measures to generalise across transformer families.

Future work: Natural extensions include decomposing the attention mechanism for complete per-layer characterisation, scaling the spectral analysis to larger models through low-rank approximation of $\mathbf{A}_x$, and

replicating the structural analysis across multiple models within each architecture family. Connecting spectral profiles to downstream applications such as targeted layer pruning or knowledge localisation could translate these findings into practical tools.

References

Arnab Barua, Mobyen Uddin Ahmed, and Shahina Begum. Mechanistic interpretability of relu neural networks through piecewise-affine mapping. *Machine Learning*, 115(1):17, 2026.

Yonatan Belinkov, Nadir Durrani, Fahim Dalvi, Hassan Sajjad, and James Glass. What do neural machine translation models learn about morphology? In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 861–872, 2017.

Ondřej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, Radu Soricut, Lucia Specia, and Aleš Tamchyna. Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pp. 12–58, 2014.

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

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 *International Conference on Learning Representations (ICLR)*, 2021.

Stefan Elfwing, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural Networks*, 107:3–11, 2018.

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. URL `https://transformer-circuits.pub/2021/framework/index.html`.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 5484–5495, 2021.

Mor Geva, Avi Caciularu, Kevin Wang, and Yoav Goldberg. Transformer feed-forward layers build predictions by promoting concepts in the vocabulary space. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 30–45, 2022.

Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (GELUs). *arXiv preprint arXiv:1606.08415*, 2016.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7B. *arXiv preprint arXiv:2310.06825*, 2023.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Anna Langedijk, Hosein Mohebbi, Gabriele Sarti, Willem Zuidema, and Jaap Jumelet. DecoderLens: Layerwise interpretation of encoder-decoder transformers. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 4764–4780, 2024.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics (TACL)*, 8:726–742, 2020.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. *arXiv preprint arXiv:1609.07843*, 2017.

Guido Montúfar, Razvan Pascanu, Kyunghyun Cho, and Yoshua Bengio. On the number of linear regions of deep neural networks. *Advances in neural information processing systems*, 27, 2014.

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

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. Technical report, OpenAI, 2019.

Prajit Ramachandran, Barret Zoph, and Quoc V. Le. Searching for activation functions. *arXiv preprint arXiv:1710.05941*, 2017.

Olivier Roy and Martin Vetterli. The effective rank: A measure of effective dimensionality. In *Proceedings of the 15th European Signal Processing Conference (EUSIPCO)*, pp. 606–610. IEEE, 2007.

Thiago Serra, Christian Tjandraatmadja, and Srikumar Ramalingam. Bounding and counting linear regions of deep neural networks. In *International conference on machine learning*, pp. 4558–4566. PMLR, 2018.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pp. 3319–3328, 2017.

Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. Multilingual translation with extensible multilingual pretraining and finetuning. *arXiv preprint arXiv:2008.00401*, 2020.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. LLaMA: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart Shieber. Investigating gender bias in language models using causal mediation analysis. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 12388–12401, 2020.

Peiyuan Zhang, Guangtao Zeng, Tianduo Wang, and Wei Lu. TinyLlama: An open-source small language model. *arXiv preprint arXiv:2401.02385*, 2024.

A Experimental Details

A.1 Model Specifications

BERT-base-uncased: A 12-layer encoder-only transformer with hidden dimension 768, intermediate dimension 3072, and 12 attention heads (Devlin et al., 2019). The activation function is GELU. The MLP uses

standard Linear layers (`intermediate.dense` for $\mathbf{W}_1$ and `output.dense` for $\mathbf{W}_2$). Post-norm architecture (layer normalisation applied after the residual addition). The prediction head applies a dense transform followed by GELU and layer normalisation before the unembedding projection.

mBART-50: A 24-layer encoder-decoder transformer (12 encoder, 12 decoder) with hidden dimension 1024, intermediate dimension 4096, and 16 attention heads (Liu et al., 2020; Tang et al., 2020). The activation function is GELU. Standard Linear layers (`fc1` for $\mathbf{W}_1$ and `fc2` for $\mathbf{W}_2$). Pre-norm architecture. The decoder employs both self-attention and cross-attention at each layer. Encoder representations are averaged across source token positions, and decoder representations are captured at each generated token position via autoregressive decoding.

GPT-2 Small: A 12-layer decoder-only transformer with hidden dimension 768, intermediate dimension 3072, and 12 attention heads (Radford et al., 2019). The activation function is GELU (`gelu_new`). The MLP uses Conv1D layers (`c_fc` for $\mathbf{W}_1$ and `c_proj` for $\mathbf{W}_2$), which store weights in transposed form relative to standard Linear layers; all weights are transposed during extraction. Pre-norm architecture. All token positions are analysed per sentence, with metrics averaged across positions.

ViT-base-patch16-224: A 12-layer encoder-only vision transformer with hidden dimension 768, intermediate dimension 3072, and 12 attention heads (Dosovitskiy et al., 2021). The activation function is GELU. Standard Linear layers with the same naming convention as BERT (`intermediate.dense` and `output.dense`). Pre-norm architecture. Input images are divided into 16×16 patches, producing 196 patch tokens plus one `[CLS]` token (197 positions total). Verification is performed at the `[CLS]` position.

TinyLlama-1.1B: A 22-layer decoder-only transformer following the LLaMA architecture, with hidden dimension 2048, intermediate dimension 5632, and 32 attention heads (Zhang et al., 2024). The activation function is SiLU. The MLP uses a gated structure with three weight matrices (`gate_proj` for $\mathbf{W}_g$, `up_proj` for $\mathbf{W}_u$, and `down_proj` for $\mathbf{W}_d$) and no bias parameters. Pre-norm architecture using RMSNorm. Verification is performed at the last token position per sample.

A.2 Datasets and Preprocessing

WikiText-103 (BERT, GPT-2, TinyLlama): The test split of WikiText-103 (Merity et al., 2017) is used. For BERT, sentences are tokenised and a content word (alphabetic, length > 2, non-subword) is randomly selected for masking; 200 masked samples are prepared with a fixed random seed (42) for reproducibility. For GPT-2 and TinyLlama, contiguous passages of 20–100 tokens are extracted, yielding 200 samples each.

WMT14 EN-DE (mBART): The test split of WMT14 English-to-German (Bojar et al., 2014) is used. Source sentences are encoded, and translations are generated autoregressively using greedy decoding. The first 200 sentence pairs are used for all experiments.

CIFAR-10 (ViT): The test split of CIFAR-10 (Krizhevsky, 2009) is used. Images (32×32) are resized to 224×224 using the standard ViT image processor. The first 200 images are used for verification.

A.3 Ablation Protocol

For each MLP block, the ablation procedure sets the block's output to zero (via a forward hook on the output projection layer) and measures the change in the target token's logit. For BERT, the target logit corresponds to the correct masked token at the `[MASK]` position. For mBART, the target logit corresponds to the next generated token at each decoder position, averaged across all positions per sentence. For GPT-2, the target logit corresponds to the next token at each position, averaged across all positions per sentence. A positive logit drop indicates that the block contributes to the correct prediction; a negative drop indicates that the block's presence is detrimental.

A.4 Computational Environment

Experiments on BERT and mBART are conducted on a Windows workstation equipped with an NVIDIA RTX 1000 Ada GPU (6.4 GB VRAM). Experiments on GPT-2, TinyLlama, and ViT are conducted on a Linux server equipped with an NVIDIA RTX A4500 GPU (21 GB VRAM). Both environments use PyTorch 2.5.1 with CUDA 12.1 and Transformers 4.40.0. The decomposition is implemented in NumPy (CPU) using SciPy for the Gaussian CDF computation, ensuring deterministic and reproducible results independent of GPU numerical behaviour.

B Verification Details

Tables 5 to 9 report per-block verification results for all five models. Each row shows the average maximum error and average mean error across all samples for one MLP block, along with the percentage of samples for which the maximum error falls below 10^{-4}.

Table 5: Per-block verification results for BERT-base (200 samples).

Block	Max Error (mean ± std)	Mean Error (mean ± std)	Pass Rate
layer_0	$8.86 \times 10^{-7} \pm 6.10 \times 10^{-7}$	$5.25 \times 10^{-8} \pm 6.71 \times 10^{-9}$	100%
layer_1	$3.52 \times 10^{-7} \pm 1.90 \times 10^{-7}$	$4.37 \times 10^{-8} \pm 5.02 \times 10^{-9}$	100%
layer_2	$2.58 \times 10^{-7} \pm 9.49 \times 10^{-8}$	$4.25 \times 10^{-8} \pm 5.73 \times 10^{-9}$	100%
layer_3	$2.30 \times 10^{-7} \pm 5.20 \times 10^{-8}$	$4.75 \times 10^{-8} \pm 5.66 \times 10^{-9}$	100%
layer_4	$2.24 \times 10^{-7} \pm 6.77 \times 10^{-8}$	$4.31 \times 10^{-8} \pm 5.04 \times 10^{-9}$	100%
layer_5	$2.03 \times 10^{-7} \pm 4.49 \times 10^{-8}$	$4.20 \times 10^{-8} \pm 5.80 \times 10^{-9}$	100%
layer_6	$2.53 \times 10^{-7} \pm 1.12 \times 10^{-7}$	$3.86 \times 10^{-8} \pm 4.59 \times 10^{-9}$	100%
layer_7	$3.11 \times 10^{-7} \pm 1.60 \times 10^{-7}$	$4.04 \times 10^{-8} \pm 4.77 \times 10^{-9}$	100%
layer_8	$2.09 \times 10^{-7} \pm 7.26 \times 10^{-8}$	$3.97 \times 10^{-8} \pm 4.55 \times 10^{-9}$	100%
layer_9	$1.95 \times 10^{-7} \pm 5.82 \times 10^{-8}$	$3.87 \times 10^{-8} \pm 5.38 \times 10^{-9}$	100%
layer_10	$2.00 \times 10^{-7} \pm 6.14 \times 10^{-8}$	$4.02 \times 10^{-8} \pm 6.09 \times 10^{-9}$	100%
layer_11	$1.98 \times 10^{-7} \pm 5.44 \times 10^{-8}$	$4.45 \times 10^{-8} \pm 7.27 \times 10^{-9}$	100%

C Gate Statistics

Tables 10 to 12 report per-block gate statistics for the three interpretability models. For each block, the soft gate vector $\mathbf{g} = \Phi(\mathbf{z})$ is computed and neurons are classified as nearly inactive ($g_i < 0.05$), nearly fully active ($g_i > 0.95$), or partially active ($0.05 \leq g_i \leq 0.95$). Values are averaged across all analysed positions and all 200 samples.

The gate vectors reveal three distinct activation patterns across architectures. In BERT, gate activity increases monotonically from 20.8% partially active neurons at layer 0 to 65.5% at layer 11, indicating progressively broader neuron engagement in later layers. In the mBART decoder, the overall trend is reversed: gate activity rises to a peak near decoder 3 (95.1%) and then decreases to 70.9% at decoder 10, reflecting increasing selectivity in the later decoder layers. In GPT-2, gate activity remains relatively flat between 68% and 83% with no systematic trend across layers.

D Full Ablation Results

Tables 13 to 15 report per-block ablation results for the three interpretability models. Each block is zeroed independently, and the average logit drop across all analysed positions is recorded. A positive drop indicates that the block contributes to the correct prediction; a negative drop indicates that removing the block improves the target logit.

Table 6: Per-block verification results for mBART-50 (200 sentences). Encoder and decoder blocks are listed separately.

Block	Max Error (mean ± std)	Mean Error (mean ± std)	Pass Rate
encoder_0	$6.60 \times 10^{-7} \pm 4.40 \times 10^{-7}$	$3.44 \times 10^{-8} \pm 4.50 \times 10^{-9}$	100%
encoder_1	$5.06 \times 10^{-6} \pm 3.83 \times 10^{-6}$	$6.19 \times 10^{-8} \pm 1.73 \times 10^{-8}$	100%
encoder_2	$6.73 \times 10^{-6} \pm 4.83 \times 10^{-6}$	$7.32 \times 10^{-8} \pm 1.80 \times 10^{-8}$	100%
encoder_3	$5.67 \times 10^{-6} \pm 4.21 \times 10^{-6}$	$7.47 \times 10^{-8} \pm 1.60 \times 10^{-8}$	100%
encoder_4	$6.52 \times 10^{-6} \pm 4.58 \times 10^{-6}$	$8.15 \times 10^{-8} \pm 1.52 \times 10^{-8}$	100%
encoder_5	$5.95 \times 10^{-6} \pm 4.63 \times 10^{-6}$	$8.95 \times 10^{-8} \pm 1.62 \times 10^{-8}$	100%
encoder_6	$4.79 \times 10^{-6} \pm 3.91 \times 10^{-6}$	$9.08 \times 10^{-8} \pm 1.39 \times 10^{-8}$	100%
encoder_7	$4.51 \times 10^{-6} \pm 3.43 \times 10^{-6}$	$9.74 \times 10^{-8} \pm 1.35 \times 10^{-8}$	100%
encoder_8	$4.98 \times 10^{-6} \pm 3.60 \times 10^{-6}$	$1.12 \times 10^{-7} \pm 1.51 \times 10^{-8}$	100%
encoder_9	$3.82 \times 10^{-6} \pm 2.32 \times 10^{-6}$	$1.04 \times 10^{-7} \pm 1.15 \times 10^{-8}$	100%
encoder_10	$1.98 \times 10^{-6} \pm 1.14 \times 10^{-6}$	$1.05 \times 10^{-7} \pm 7.03 \times 10^{-9}$	100%
encoder_11	$7.06 \times 10^{-7} \pm 3.41 \times 10^{-7}$	$7.82 \times 10^{-8} \pm 6.43 \times 10^{-9}$	100%
decoder_0	$4.46 \times 10^{-6} \pm 3.35 \times 10^{-6}$	$3.56 \times 10^{-8} \pm 6.25 \times 10^{-9}$	100%
decoder_1	$7.04 \times 10^{-6} \pm 5.57 \times 10^{-6}$	$2.48 \times 10^{-8} \pm 7.73 \times 10^{-9}$	100%
decoder_2	$6.56 \times 10^{-6} \pm 4.15 \times 10^{-6}$	$2.54 \times 10^{-8} \pm 5.56 \times 10^{-9}$	100%
decoder_3	$5.98 \times 10^{-6} \pm 4.62 \times 10^{-6}$	$3.14 \times 10^{-8} \pm 6.52 \times 10^{-9}$	100%
decoder_4	$2.48 \times 10^{-7} \pm 1.31 \times 10^{-7}$	$3.09 \times 10^{-8} \pm 1.36 \times 10^{-9}$	100%
decoder_5	$2.29 \times 10^{-7} \pm 9.09 \times 10^{-8}$	$3.84 \times 10^{-8} \pm 1.39 \times 10^{-9}$	100%
decoder_6	$2.31 \times 10^{-7} \pm 7.32 \times 10^{-8}$	$4.28 \times 10^{-8} \pm 1.55 \times 10^{-9}$	100%
decoder_7	$2.67 \times 10^{-7} \pm 1.04 \times 10^{-7}$	$4.52 \times 10^{-8} \pm 1.67 \times 10^{-9}$	100%
decoder_8	$3.35 \times 10^{-7} \pm 1.61 \times 10^{-7}$	$4.64 \times 10^{-8} \pm 1.59 \times 10^{-9}$	100%
decoder_9	$4.29 \times 10^{-7} \pm 2.19 \times 10^{-7}$	$4.89 \times 10^{-8} \pm 2.01 \times 10^{-9}$	100%
decoder_10	$3.36 \times 10^{-7} \pm 1.06 \times 10^{-7}$	$5.66 \times 10^{-8} \pm 2.03 \times 10^{-9}$	100%
decoder_11	$7.11 \times 10^{-7} \pm 3.34 \times 10^{-7}$	$9.08 \times 10^{-8} \pm 5.96 \times 10^{-9}$	100%

Table 7: Per-block verification results for GPT-2 Small (200 samples).

Block	Max Error (mean ± std)	Mean Error (mean ± std)	Pass Rate
layer_0	$2.83 \times 10^{-6} \pm 1.11 \times 10^{-6}$	$2.27 \times 10^{-7} \pm 4.97 \times 10^{-8}$	100%
layer_1	$2.00 \times 10^{-6} \pm 8.33 \times 10^{-7}$	$2.06 \times 10^{-7} \pm 4.47 \times 10^{-8}$	100%
layer_2	$2.25 \times 10^{-6} \pm 1.29 \times 10^{-6}$	$2.24 \times 10^{-7} \pm 3.54 \times 10^{-8}$	100%
layer_3	$2.28 \times 10^{-6} \pm 1.48 \times 10^{-6}$	$2.19 \times 10^{-7} \pm 3.02 \times 10^{-8}$	100%
layer_4	$2.35 \times 10^{-6} \pm 1.43 \times 10^{-6}$	$2.40 \times 10^{-7} \pm 2.68 \times 10^{-8}$	100%
layer_5	$2.06 \times 10^{-6} \pm 1.11 \times 10^{-6}$	$2.63 \times 10^{-7} \pm 2.80 \times 10^{-8}$	100%
layer_6	$2.25 \times 10^{-6} \pm 1.17 \times 10^{-6}$	$3.03 \times 10^{-7} \pm 3.40 \times 10^{-8}$	100%
layer_7	$2.92 \times 10^{-6} \pm 1.88 \times 10^{-6}$	$3.39 \times 10^{-7} \pm 4.23 \times 10^{-8}$	100%
layer_8	$2.87 \times 10^{-6} \pm 1.34 \times 10^{-6}$	$4.09 \times 10^{-7} \pm 5.22 \times 10^{-8}$	100%
layer_9	$3.82 \times 10^{-6} \pm 2.12 \times 10^{-6}$	$4.92 \times 10^{-7} \pm 6.90 \times 10^{-8}$	100%
layer_10	$5.55 \times 10^{-6} \pm 3.02 \times 10^{-6}$	$6.31 \times 10^{-7} \pm 1.14 \times 10^{-7}$	100%
layer_11	$8.29 \times 10^{-6} \pm 5.56 \times 10^{-6}$	$7.38 \times 10^{-7} \pm 1.46 \times 10^{-7}$	100%

E Baseline Comparison

Tables 16 to 18 report the full baseline comparison for each model, showing Spearman rank correlation (ρ) and p-values between each importance measure and the ablation ground truth.

Table 8: Per-block verification results for ViT-base (200 images).

Block	Max Error (mean ± std)	Mean Error (mean ± std)	Pass Rate
layer_0	$3.99 \times 10^{-7} \pm 2.48 \times 10^{-7}$	$2.16 \times 10^{-8} \pm 1.62 \times 10^{-9}$	100%
layer_1	$5.99 \times 10^{-7} \pm 3.30 \times 10^{-7}$	$2.51 \times 10^{-8} \pm 2.04 \times 10^{-9}$	100%
layer_2	$2.44 \times 10^{-7} \pm 1.21 \times 10^{-7}$	$2.68 \times 10^{-8} \pm 2.89 \times 10^{-9}$	100%
layer_3	$1.24 \times 10^{-7} \pm 2.33 \times 10^{-8}$	$2.85 \times 10^{-8} \pm 2.98 \times 10^{-9}$	100%
layer_4	$1.97 \times 10^{-7} \pm 3.83 \times 10^{-8}$	$4.48 \times 10^{-8} \pm 5.65 \times 10^{-9}$	100%
layer_5	$2.23 \times 10^{-7} \pm 4.17 \times 10^{-8}$	$5.13 \times 10^{-8} \pm 6.25 \times 10^{-9}$	100%
layer_6	$2.83 \times 10^{-7} \pm 7.11 \times 10^{-8}$	$6.03 \times 10^{-8} \pm 7.00 \times 10^{-9}$	100%
layer_7	$3.93 \times 10^{-7} \pm 1.61 \times 10^{-7}$	$6.29 \times 10^{-8} \pm 6.51 \times 10^{-9}$	100%
layer_8	$3.50 \times 10^{-7} \pm 9.48 \times 10^{-8}$	$7.26 \times 10^{-8} \pm 8.23 \times 10^{-9}$	100%
layer_9	$3.25 \times 10^{-7} \pm 1.18 \times 10^{-7}$	$6.66 \times 10^{-8} \pm 1.03 \times 10^{-8}$	100%
layer_10	$5.99 \times 10^{-7} \pm 1.64 \times 10^{-7}$	$1.34 \times 10^{-7} \pm 3.15 \times 10^{-8}$	100%
layer_11	$1.49 \times 10^{-6} \pm 5.12 \times 10^{-7}$	$2.64 \times 10^{-7} \pm 4.67 \times 10^{-8}$	100%

Table 9: Per-block verification results for TinyLlama-1.1B (200 samples, gated MLP with SiLU).

Block	Max Error (mean ± std)	Mean Error (mean ± std)	Pass Rate
layer_0	$1.65 \times 10^{-8} \pm 9.48 \times 10^{-9}$	$1.26 \times 10^{-9} \pm 5.01 \times 10^{-10}$	100%
layer_1	$1.52 \times 10^{-8} \pm 5.81 \times 10^{-9}$	$2.23 \times 10^{-9} \pm 9.10 \times 10^{-10}$	100%
layer_2	$2.49 \times 10^{-8} \pm 1.16 \times 10^{-8}$	$3.30 \times 10^{-9} \pm 1.23 \times 10^{-9}$	100%
layer_3	$3.16 \times 10^{-8} \pm 1.31 \times 10^{-8}$	$4.85 \times 10^{-9} \pm 8.82 \times 10^{-10}$	100%
layer_4	$3.19 \times 10^{-8} \pm 8.82 \times 10^{-9}$	$6.21 \times 10^{-9} \pm 1.01 \times 10^{-9}$	100%
layer_5	$4.09 \times 10^{-8} \pm 1.09 \times 10^{-8}$	$7.85 \times 10^{-9} \pm 1.22 \times 10^{-9}$	100%
layer_6	$5.81 \times 10^{-8} \pm 2.20 \times 10^{-8}$	$9.59 \times 10^{-9} \pm 1.41 \times 10^{-9}$	100%
layer_7	$5.03 \times 10^{-6} \pm 1.43 \times 10^{-5}$	$2.77 \times 10^{-8} \pm 3.93 \times 10^{-8}$	100%
layer_8	$7.38 \times 10^{-8} \pm 2.39 \times 10^{-8}$	$1.21 \times 10^{-8} \pm 3.78 \times 10^{-9}$	100%
layer_9	$7.30 \times 10^{-8} \pm 2.64 \times 10^{-8}$	$1.41 \times 10^{-8} \pm 4.59 \times 10^{-9}$	100%
layer_10	$8.46 \times 10^{-8} \pm 3.77 \times 10^{-8}$	$1.46 \times 10^{-8} \pm 4.80 \times 10^{-9}$	100%
layer_11	$8.66 \times 10^{-8} \pm 3.06 \times 10^{-8}$	$1.65 \times 10^{-8} \pm 5.31 \times 10^{-9}$	100%
layer_12	$1.06 \times 10^{-7} \pm 4.62 \times 10^{-8}$	$1.94 \times 10^{-8} \pm 6.41 \times 10^{-9}$	100%
layer_13	$1.23 \times 10^{-7} \pm 4.95 \times 10^{-8}$	$2.22 \times 10^{-8} \pm 7.37 \times 10^{-9}$	100%
layer_14	$1.35 \times 10^{-7} \pm 5.57 \times 10^{-8}$	$2.45 \times 10^{-8} \pm 8.17 \times 10^{-9}$	100%
layer_15	$1.81 \times 10^{-7} \pm 8.59 \times 10^{-8}$	$2.99 \times 10^{-8} \pm 9.97 \times 10^{-9}$	100%
layer_16	$2.40 \times 10^{-7} \pm 1.12 \times 10^{-7}$	$3.84 \times 10^{-8} \pm 1.31 \times 10^{-8}$	100%
layer_17	$2.82 \times 10^{-7} \pm 1.23 \times 10^{-7}$	$4.20 \times 10^{-8} \pm 1.46 \times 10^{-8}$	100%
layer_18	$3.38 \times 10^{-7} \pm 1.44 \times 10^{-7}$	$5.33 \times 10^{-8} \pm 1.84 \times 10^{-8}$	100%
layer_19	$4.52 \times 10^{-7} \pm 1.70 \times 10^{-7}$	$6.92 \times 10^{-8} \pm 2.28 \times 10^{-8}$	100%
layer_20	$8.48 \times 10^{-7} \pm 3.73 \times 10^{-7}$	$9.38 \times 10^{-8} \pm 2.23 \times 10^{-8}$	100%
layer_21	$8.25 \times 10^{-6} \pm 1.98 \times 10^{-5}$	$1.41 \times 10^{-7} \pm 3.64 \times 10^{-8}$	99%

F Consistency and Stability

This section reports cross-sample consistency of block rankings (Experiment 6) and determinism of the decomposition (Experiment 7). The decomposition produces identical rankings and values on repeated computation of the same input (Spearman $\rho = 1.000$ across all models). Block importance rankings are consistent across samples, with the top-ranked block maintaining its position in over 90% of samples on all three models.

Table 10: Per-block gate statistics for BERT-base (200 samples).

Block	Gate Mean	Off	On	Between	% Between
layer_0	0.064	2411	21	639	20.8
layer_1	0.087	2078	9	983	32.0
layer_2	0.086	2097	7	967	31.5
layer_3	0.101	1945	8	1118	36.4
layer_4	0.103	1889	5	1176	38.3
layer_5	0.112	1774	6	1291	42.0
layer_6	0.106	1828	4	1239	40.3
layer_7	0.123	1626	6	1439	46.9
layer_8	0.136	1444	6	1621	52.8
layer_9	0.138	1322	4	1744	56.8
layer_10	0.148	1160	3	1908	62.1
layer_11	0.150	1057	2	2012	65.5

Table 11: Per-block gate statistics for mBART-50 (200 samples).

Block	Gate Mean	Off	On	Between	% Between
encoder_0	0.239	220	5	3870	94.5
encoder_1	0.271	274	7	3814	93.1
encoder_2	0.342	164	13	3918	95.7
encoder_3	0.408	69	13	4013	98.0
encoder_4	0.427	80	11	4004	97.8
encoder_5	0.417	86	9	4000	97.7
encoder_6	0.380	253	13	3829	93.5
encoder_7	0.337	464	12	3620	88.4
encoder_8	0.326	543	11	3541	86.5
encoder_9	0.338	471	8	3615	88.3
encoder_10	0.328	566	5	3524	86.0
encoder_11	0.333	303	4	3788	92.5
decoder_0	0.313	286	3	3805	92.9
decoder_1	0.361	406	8	3681	89.9
decoder_2	0.342	237	4	3854	94.1
decoder_3	0.323	199	0	3896	95.1
decoder_4	0.331	214	1	3879	94.7
decoder_5	0.312	351	6	3737	91.3
decoder_6	0.334	360	12	3723	90.9
decoder_7	0.352	399	19	3677	89.8
decoder_8	0.391	491	60	3544	86.5
decoder_9	0.380	709	167	3219	78.6
decoder_10	0.374	949	244	2902	70.9
decoder_11	0.366	723	153	3219	78.6

G Computational Efficiency

This section reports wall-clock timing comparisons between the proposed decomposition (fast path), gradient-based attribution, and full ablation for each model.

H Full $\mathbf{A}_x$ Analysis Results

Tables 28 to 30 report the complete per-block $\mathbf{A}_x$ spectral metrics (Frobenius norm, spectral norm, nuclear norm, effective rank, stable rank, and trace) with standard deviations across 200 samples.

Table 12: Per-block gate statistics for GPT-2 Small (200 samples).

Block	Gate Mean	Off	On	Between	% Between
layer_0	0.218	737	7	2327	75.8
layer_1	0.319	511	2	2557	83.2
layer_2	0.239	881	2	2187	71.2
layer_3	0.207	965	2	2103	68.5
layer_4	0.239	750	2	2319	75.5
layer_5	0.230	815	3	2253	73.4
layer_6	0.240	702	3	2366	77.0
layer_7	0.238	652	3	2415	78.6
layer_8	0.233	656	4	2410	78.5
layer_9	0.229	657	5	2408	78.4
layer_10	0.243	612	12	2447	79.7
layer_11	0.264	532	16	2522	82.1

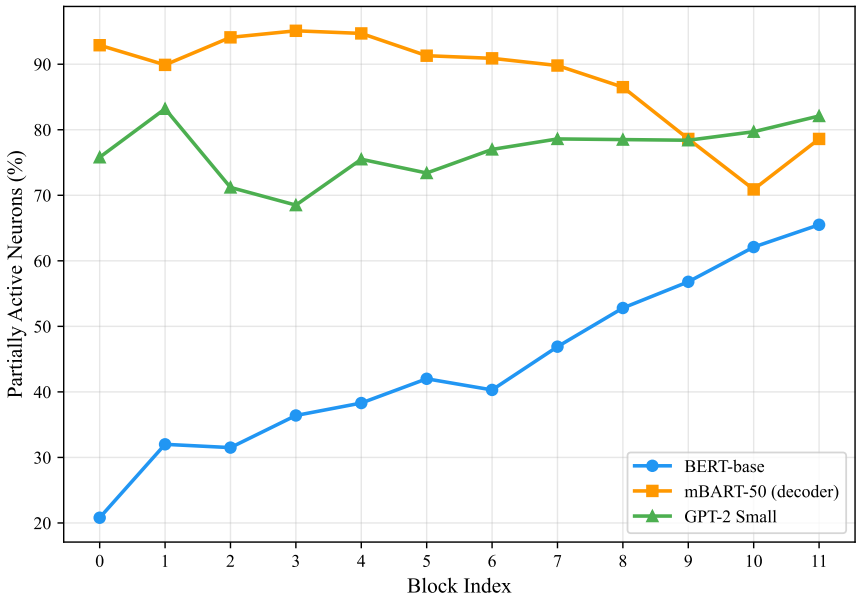


Figure 2: Percentage of partially active neurons ($0.05 \leq g_i \leq 0.95$) per block for the three interpretability models.

Table 13: Per-block ablation results for BERT-base. Blocks sorted by index.

Block	Avg Logit Drop (mean)	Std
layer_0	0.3054	0.6489
layer_1	0.2222	0.7118
layer_2	0.3294	0.6792
layer_3	0.3839	0.8070
layer_4	0.6196	0.8643
layer_5	0.6666	0.8581
layer_6	0.4797	0.7800
layer_7	0.7058	0.9011
layer_8	0.7917	0.8802
layer_9	0.8797	1.1670
layer_10	1.0082	1.3343
layer_11	0.8506	1.4175

Table 14: Per-block ablation results for mBART-50. Blocks sorted by index.

Block	Avg Logit Drop (mean)	Std
encoder_0	0.0400	0.0522
encoder_1	0.0315	0.0451
encoder_2	0.0230	0.0359
encoder_3	0.0145	0.0332
encoder_4	0.0186	0.0375
encoder_5	0.0097	0.0471
encoder_6	0.0057	0.0508
encoder_7	0.0063	0.0646
encoder_8	0.0328	0.0604
encoder_9	0.0077	0.0523
encoder_10	0.0142	0.0552
encoder_11	0.0125	0.0527
decoder_0	−0.0746	0.0877
decoder_1	−0.0573	0.0584
decoder_2	−0.0397	0.0510
decoder_3	−0.0241	0.0560
decoder_4	−0.0448	0.0612
decoder_5	0.0043	0.0690
decoder_6	0.0843	0.0822
decoder_7	0.1738	0.0869
decoder_8	0.3677	0.1058
decoder_9	0.5426	0.1214
decoder_10	0.5260	0.1997
decoder_11	−0.2266	0.2499

Table 15: Per-block ablation results for GPT-2 Small. Blocks sorted by index.

Block	Avg Logit Drop (mean)	Std
layer_0	−4.7849	9.2576
layer_1	2.2043	1.2729
layer_2	1.2487	1.2614
layer_3	2.4687	1.2821
layer_4	2.0164	1.3891
layer_5	1.9648	1.3888
layer_6	1.3449	1.3162
layer_7	0.7816	1.5698
layer_8	1.0935	1.8224
layer_9	−0.3299	1.9991
layer_10	−16.4394	3.1735
layer_11	−10.5052	2.4421

Table 16: Baseline comparison for BERT-base and Spearman correlation with ablation.

Method	ρ (all)	p-value	ρ (excl)	p-value
L2 Norm	−0.7972	1.90×10^{-3}	−0.7727	5.30×10^{-3}
|DLA|	−0.8462	5.20×10^{-4}	−0.8636	6.12×10^{-4}
Gradient-based	0.9510	2.05×10^{-6}	0.9636	1.86×10^{-6}
Random	-9.00×10^{-4}	9.98×10^{-1}	−0.0175	9.59×10^{-1}

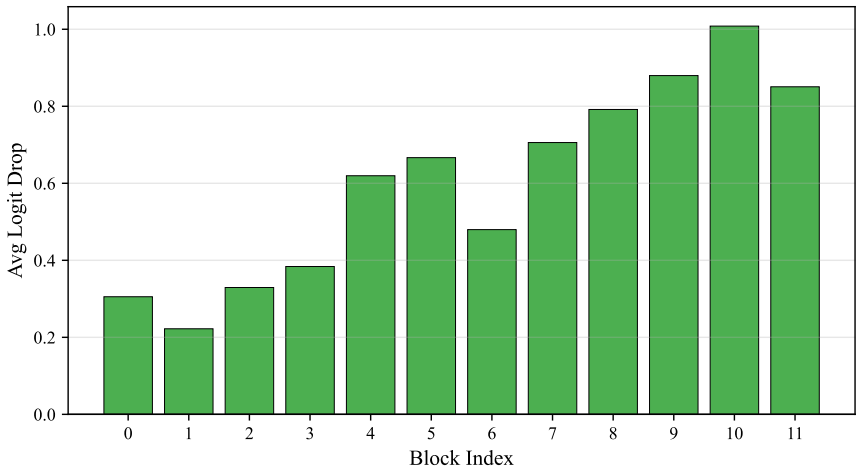


Figure 3: Ablation profile for BERT-base. All blocks contribute positively, with importance increasing with depth.

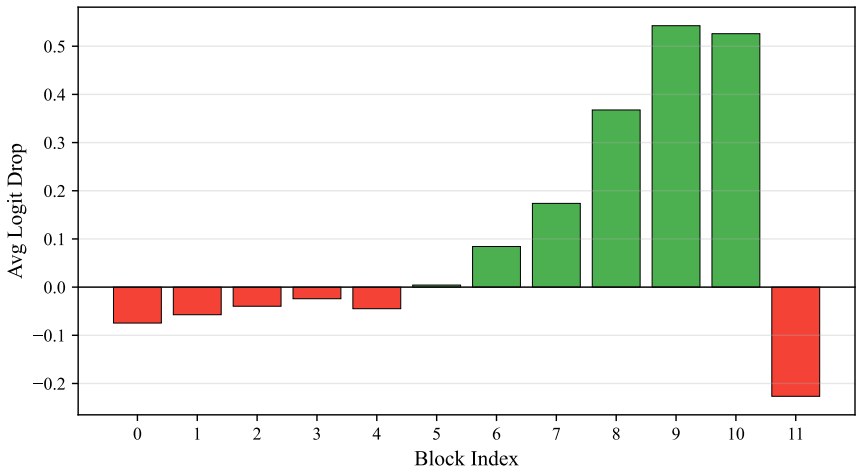


Figure 4: Ablation profile for mBART-50 (decoder blocks). Importance concentrates in layers 8–10. Decoder 11 produces a negative logit drop.

Table 17: Baseline comparison for mBART-50. Spearman correlation with ablation.

Method	ρ **(all)**	p**-value**	ρ **(excl)**	p**-value**
L2 Norm	0.4476	1.45×10^{-1}	0.7909	3.75×10^{-3}
|DLA|	0.1608	6.18×10^{-1}	0.4182	2.01×10^{-1}
Gradient-based	−0.6853	1.39×10^{-2}	−0.5909	5.56×10^{-2}
Random	0.0018	9.96×10^{-1}	−0.0143	9.67×10^{-1}

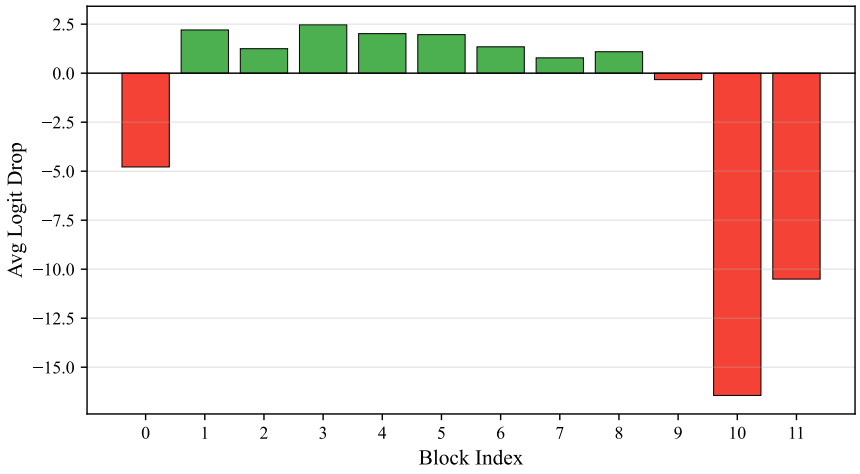


Figure 5: Ablation profile for GPT-2 Small. Middle layers produce the largest positive drops. Late layers (10, 11) produce large negative drops.

Table 18: Baseline comparison for GPT-2 Small. Spearman correlation with ablation.

Method	ρ **(all)**	p**-value**	ρ **(excl)**	p**-value**
L2 Norm	-0.8811	1.53×10^{-4}	-0.8545	8.08×10^{-4}
|DLA|	-0.8741	2.01×10^{-4}	-0.8455	1.04×10^{-3}
Gradient-based	0.8601	3.32×10^{-4}	0.8273	1.68×10^{-3}
Logit Lens	-0.9231	1.86×10^{-5}	-0.9273	3.97×10^{-5}
Activation Patching	0.6503	2.20×10^{-2}	0.5636	7.10×10^{-2}
Random	0.0110	9.73×10^{-1}	0.0088	9.80×10^{-1}

Table 19: Consistency results for BERT-base.

Block	**Mean Rank**	**Std**	**Min**	**Max**	**Rank 1**	**Top-3**
layer_0	1.0000	0.0700	1.0000	2.0000	199.00	200.00
layer_1	2.6200	1.4300	1.0000	11.0000	1.0000	170.00
layer_3	4.5700	2.0100	2.0000	12.0000	0.0000	81.0000
layer_2	6.3300	2.5800	2.0000	12.0000	0.0000	27.0000
layer_7	6.5100	2.4500	2.0000	12.0000	0.0000	26.0000
layer_8	6.6400	2.5300	2.0000	12.0000	0.0000	23.0000
layer_4	7.1700	2.5200	2.0000	12.0000	0.0000	17.0000
layer_5	7.2200	2.8900	2.0000	12.0000	0.0000	22.0000
layer_9	7.5000	2.4300	2.0000	12.0000	0.0000	14.0000
layer_6	8.7700	2.4400	2.0000	12.0000	0.0000	9.0000
layer_10	9.0000	2.5400	2.0000	12.0000	0.0000	7.0000
layer_11	10.6500	2.2800	2.0000	12.0000	0.0000	4.0000

Table 20: Consistency results for mBART-50.

Block	Mean Rank	Std	Min Rank	Max Rank	Rank 1 Count	Top-3 Count
decoder_10	1.1400	0.3400	1.0000	2.0000	173.00	200.00
decoder_11	1.8600	0.3400	1.0000	2.0000	27.0000	200.00
decoder_9	3.0000	0.0700	3.0000	4.0000	0.0000	199.00
decoder_8	4.0300	0.2000	4.0000	6.0000	0.0000	0.0000
decoder_7	5.1800	0.4900	5.0000	8.0000	0.0000	0.0000
decoder_6	6.4100	0.7800	6.0000	10.0000	0.0000	0.0000
decoder_5	7.8500	1.1000	7.0000	11.0000	0.0000	0.0000
decoder_1	8.0300	2.1500	3.0000	12.0000	0.0000	1.0000
decoder_0	8.8700	1.1300	5.0000	11.0000	0.0000	0.0000
decoder_4	9.6400	1.5300	8.0000	12.0000	0.0000	0.0000
decoder_3	10.7900	0.9000	9.0000	12.0000	0.0000	0.0000
decoder_2	11.1900	1.1100	7.0000	12.0000	0.0000	0.0000

Table 21: Consistency results for GPT-2 Small.

Block	Mean Rank	Std	Min	Max	Rank 1	Top-3
layer_11	1.1300	0.3600	1.0000	3.0000	176.00	200.00
layer_10	1.9100	0.3200	1.0000	3.0000	20.0000	200.00
layer_9	3.2600	0.4400	3.0000	4.0000	0.0000	148.00
layer_2	3.7000	0.5700	1.0000	4.0000	4.0000	52.0000
layer_8	5.2500	0.4500	5.0000	7.0000	0.0000	0.0000
layer_0	5.7500	0.4300	5.0000	6.0000	0.0000	0.0000
layer_7	7.0200	0.1400	7.0000	8.0000	0.0000	0.0000
layer_6	8.0400	0.1800	8.0000	9.0000	0.0000	0.0000
layer_5	9.1000	0.3700	9.0000	12.0000	0.0000	0.0000
layer_1	10.3900	1.0400	6.0000	12.0000	0.0000	0.0000
layer_4	11.1200	0.7900	10.0000	12.0000	0.0000	0.0000
layer_3	11.3200	0.6500	10.0000	12.0000	0.0000	0.0000

Table 22: Stability results for BERT-base.

Metric	Value
Spearman rho (mean)	1.0000
Top-1 changes	0/49
Mean CV	0.0000
Verdict	PERFECTLY DETERMINISTIC

Table 23: Stability results for mBART-50.

Metric	Value
Number of runs	50.0000
Spearman rho (mean)	1.0000
Spearman rho (min)	1.0000
Spearman rho (max)	1.0000
Top-1 changes	0/49
Mean CV	0.0000
Max CV	0.0000
Verdict	PERFECTLY DETERMINISTIC
Block	CV
decoder_0	0.0000
decoder_1	0.0000
decoder_2	0.0000
decoder_3	0.0000
decoder_4	0.0000
decoder_5	0.0000
decoder_6	0.0000
decoder_7	0.0000
decoder_8	0.0000
decoder_9	0.0000
decoder_10	0.0000
decoder_11	0.0000

Table 24: Stability results for GPT-2 Small.

Metric	Value
Spearman rho (mean)	1.0000
Top-1 changes	0/49
Mean CV	0.0000
Verdict	PERFECTLY DETERMINISTIC

Table 25: Computational efficiency for BERT-base.

Method	Avg Time (s)	Relative
Proposed (L2+DLA)	0.1860	1.0x
Gradient-based	0.0650	0.4x
Ablation	0.2860	1.5x

Table 26: Computational efficiency for mBART-50.

Method	Avg Time (s)	Std (s)	Relative Speed	Forward Passes
Proposed (L2+DLA)	9.2330	3.3280	1.0x	1 per token
Gradient-based	39.8640	15.2840	4.3x	1 fwd+bwd per token
Ablation (24 blocks)	50.4290	22.7770	5.5x	24 per token
LN-aware	4.1230	1.7060	0.4x	1 per token + 12 LN

Table 27: Computational efficiency for GPT-2 Small.

Method	Avg Time (s)	Relative
Proposed (L2+DLA)	1.3870	1.0x
Gradient-based	0.0270	0.0x
Ablation	0.0790	0.1x

Table 28: Per-block $\mathbf{A}_x$ spectral metrics for BERT-base (mean ± std across 200 samples).

Block	Frob Norm	Spectral Norm	Nuclear Norm	Eff Rank	Stable Rank	Trace
layer_0	12.18 ± 0.67	7.95 ± 0.65	133.9 ± 10.0	275.6 ± 16.2	2.37 ± 0.21	21.56 ± 3.34
layer_1	11.80 ± 0.76	5.49 ± 0.51	177.2 ± 14.5	377.1 ± 11.7	4.69 ± 0.62	10.48 ± 3.36
layer_2	11.61 ± 0.95	4.71 ± 0.44	179.6 ± 18.9	380.4 ± 13.1	6.15 ± 0.92	9.27 ± 4.37
layer_3	12.91 ± 0.96	4.85 ± 0.47	210.6 ± 18.6	403.2 ± 8.9	7.21 ± 1.12	5.35 ± 3.84
layer_4	12.33 ± 0.98	4.24 ± 0.52	206.3 ± 18.1	413.5 ± 10.4	8.63 ± 1.23	2.61 ± 4.79
layer_5	12.45 ± 1.34	4.01 ± 0.66	213.6 ± 24.3	426.2 ± 11.1	9.92 ± 1.64	-5.55 ± 5.16
layer_6	11.85 ± 1.17	3.65 ± 0.54	201.7 ± 21.8	420.5 ± 11.1	10.77 ± 1.72	-9.84 ± 4.36
layer_7	12.28 ± 1.01	3.38 ± 0.36	219.1 ± 20.4	445.3 ± 10.3	13.34 ± 1.73	6.87 ± 6.29
layer_8	12.71 ± 0.96	3.12 ± 0.24	237.6 ± 20.4	470.3 ± 10.6	16.71 ± 1.78	13.31 ± 5.94
layer_9	12.89 ± 0.99	2.94 ± 0.33	248.1 ± 20.8	485.2 ± 10.6	19.60 ± 2.99	14.96 ± 5.38
layer_10	12.84 ± 1.30	2.72 ± 0.19	252.6 ± 27.0	497.2 ± 10.5	22.24 ± 2.51	-9.38 ± 2.66
layer_11	11.97 ± 1.66	1.85 ± 0.19	235.5 ± 33.7	486.3 ± 7.9	41.66 ± 5.05	-29.41 ± 4.31

Table 29: Per-block $\mathbf{A}_x$ spectral metrics for mBART-50 (decoder) (mean ± std across 200 samples).

Block	Frob Norm	Spectral Norm	Nuclear Norm	Eff Rank	Stable Rank	Trace
decoder_0	42.94 ± 1.31	13.19 ± 1.59	965.5 ± 15.7	710.0 ± 3.6	15.22 ± 0.34	-162.3 ± 26.5
decoder_1	49.35 ± 3.39	19.82 ± 3.89	1052.9 ± 12.9	732.8 ± 6.4	11.05 ± 0.36	-596.6 ± 20.2
decoder_2	48.38 ± 2.61	17.91 ± 3.16	1065.1 ± 10.7	723.3 ± 5.4	11.96 ± 0.68	-655.8 ± 17.0
decoder_3	50.65 ± 2.12	17.33 ± 2.73	1125.2 ± 10.5	712.6 ± 3.7	13.41 ± 0.66	-697.1 ± 14.3
decoder_4	51.07 ± 0.60	13.55 ± 0.53	1201.5 ± 12.4	714.7 ± 1.1	14.81 ± 0.75	-761.6 ± 13.0
decoder_5	52.73 ± 0.60	14.40 ± 0.65	1233.9 ± 10.9	711.3 ± 1.5	14.20 ± 0.83	-796.1 ± 20.6
decoder_6	59.61 ± 0.66	15.77 ± 0.65	1409.2 ± 12.0	719.6 ± 2.2	15.00 ± 0.80	-956.5 ± 20.3
decoder_7	67.06 ± 0.84	18.10 ± 0.57	1599.3 ± 18.7	728.5 ± 1.6	14.17 ± 0.65	-1120.5 ± 18.0
decoder_8	77.62 ± 0.86	25.99 ± 0.76	1820.1 ± 20.3	720.9 ± 1.6	9.24 ± 0.51	-1273.4 ± 18.6
decoder_9	84.88 ± 1.26	33.78 ± 0.84	1900.0 ± 32.4	690.3 ± 1.5	6.43 ± 0.32	-1230.4 ± 23.1
decoder_10	94.80 ± 1.16	48.45 ± 0.99	1943.6 ± 29.9	654.5 ± 1.3	3.91 ± 0.15	-1119.9 ± 22.6
decoder_11	119.8 ± 1.3	82.72 ± 1.80	1974.6 ± 19.6	581.2 ± 2.9	2.12 ± 0.06	-1056.6 ± 12.6

Table 30: Per-block $\mathbf{A}_x$ spectral metrics for GPT-2 Small (mean ± std across 200 samples).

Block	Frob Norm	Spectral Norm	Nuclear Norm	Eff Rank	Stable Rank	Trace
layer_0	157.5 ± 4.2	34.72 ± 0.58	3120.0 ± 96.9	501.1 ± 2.5	22.26 ± 1.14	-232.6 ± 23.4
layer_1	161.2 ± 1.9	45.91 ± 2.38	3497.5 ± 28.2	595.0 ± 2.5	15.28 ± 0.30	-2954.2 ± 42.7
layer_2	147.9 ± 3.2	50.05 ± 3.45	2882.8 ± 29.9	532.4 ± 2.0	11.17 ± 0.38	-2017.9 ± 33.7
layer_3	127.3 ± 1.5	51.79 ± 1.54	2383.8 ± 23.7	496.6 ± 1.0	6.55 ± 0.25	-1281.3 ± 19.8
layer_4	138.2 ± 1.6	52.30 ± 1.21	2620.8 ± 30.8	498.6 ± 1.1	7.21 ± 0.25	-1480.9 ± 23.2
layer_5	139.3 ± 1.9	53.47 ± 1.22	2645.1 ± 40.9	498.9 ± 1.7	6.94 ± 0.23	-1411.7 ± 26.5
layer_6	154.3 ± 2.5	49.76 ± 1.12	3009.5 ± 55.8	507.1 ± 1.8	9.78 ± 0.31	-1597.9 ± 36.0
layer_7	170.3 ± 2.3	56.05 ± 1.33	3322.5 ± 50.1	507.3 ± 1.2	9.39 ± 0.36	-1531.6 ± 40.1
layer_8	192.9 ± 3.0	58.21 ± 1.39	3769.0 ± 66.5	506.3 ± 1.5	11.19 ± 0.51	-1528.4 ± 36.1
layer_9	221.7 ± 3.1	71.96 ± 1.73	4295.4 ± 67.9	502.9 ± 1.3	9.75 ± 0.51	-1442.0 ± 31.6
layer_10	282.6 ± 3.1	129.2 ± 3.4	5186.5 ± 72.9	496.5 ± 1.2	4.94 ± 0.27	-1583.3 ± 33.3
layer_11	374.9 ± 6.2	215.0 ± 6.3	6008.0 ± 145.1	469.2 ± 2.6	3.09 ± 0.17	-1537.9 ± 37.8

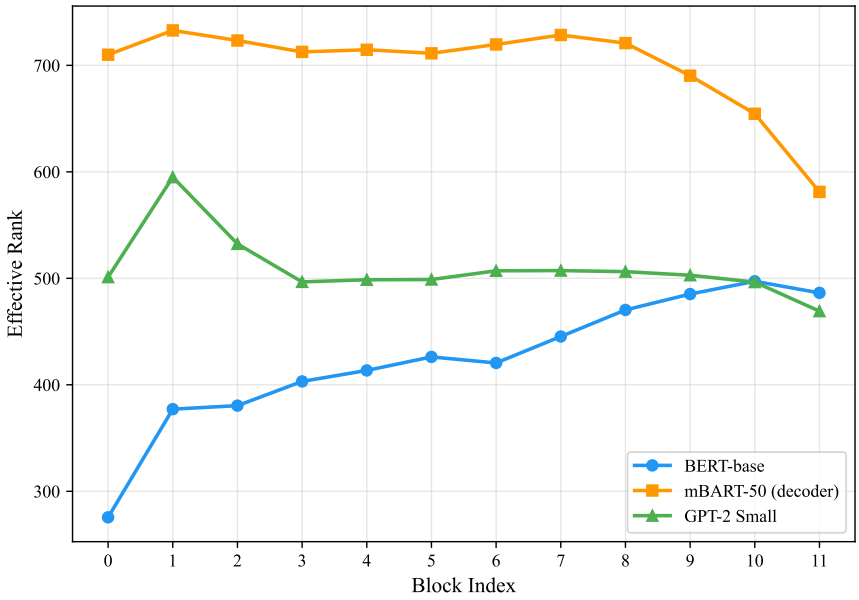


Figure 6: Effective rank of $\mathbf{A}_x$ across layers (raw values). BERT increases from 276 to 497. mBART decoder and GPT-2 remain approximately flat.

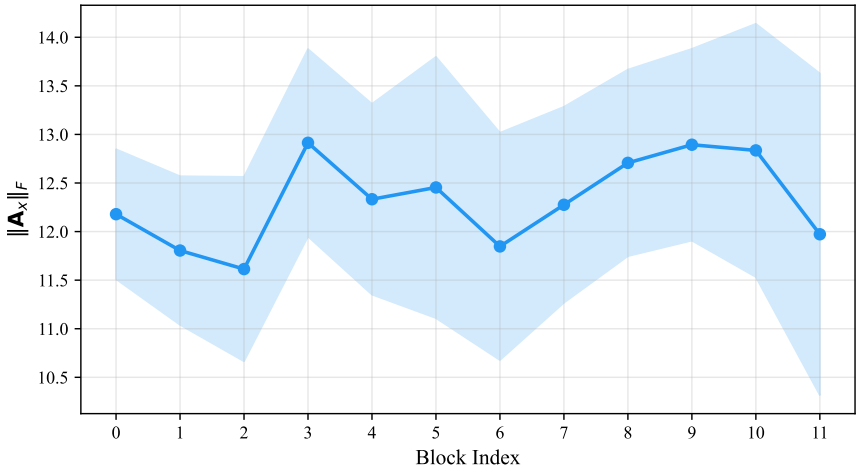


Figure 7: Frobenius norm of $\mathbf{A}_x$ across layers for BERT-base. The norm remains approximately flat (≈ 12) across all layers.

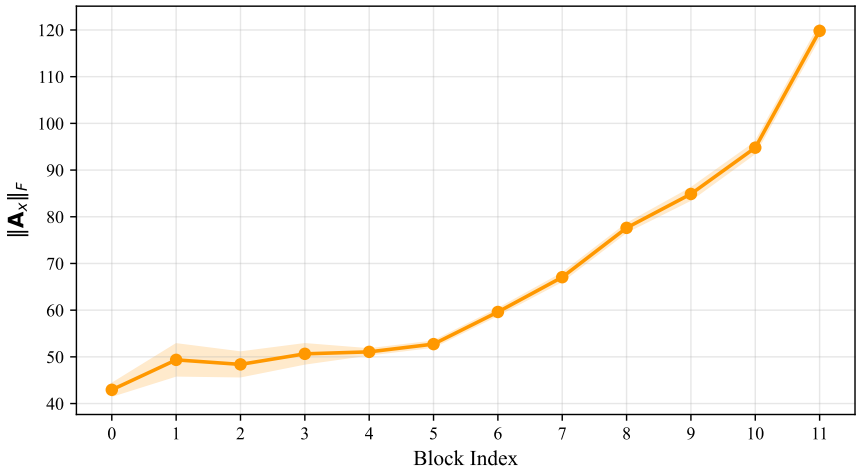


Figure 8: Frobenius norm of $\mathbf{A}_x$ across layers for mBART-50 (decoder). The norm increases monotonically from 42.9 to 94.8.

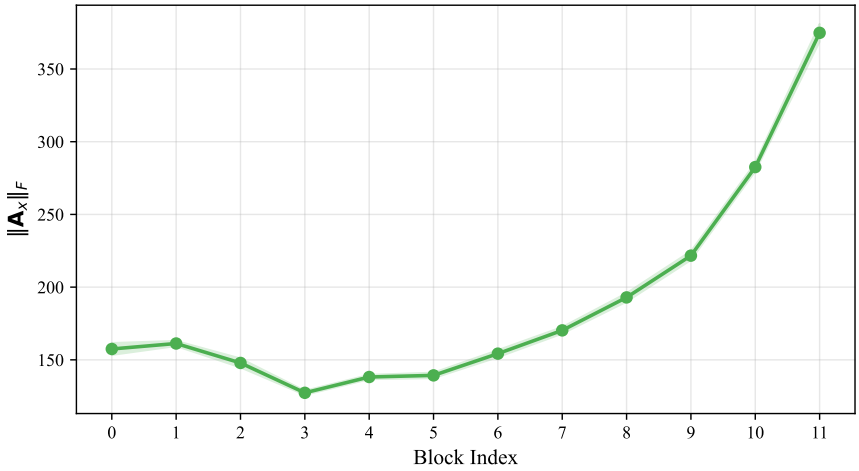


Figure 9: Frobenius norm of $\mathbf{A}_x$ across layers for GPT-2 Small. The norm increases from 127 to 375, with the steepest rise in the final two layers.