Beyond Components: Singular Vector-Based Interpretability of Transformer Circuits

Areeb Ahmad* **Abhinav Joshi*** **Ashutosh Modi**
Indian Institute of Technology Kanpur (IIT Kanpur)
{ajoshi,areeb,ashutoshm}@cse.iitk.ac.in

Abstract

Transformer-based language models exhibit complex and distributed behavior, yet their internal computations remain poorly understood. Existing mechanistic interpretability methods typically treat attention heads and multilayer perceptron layers (MLPs) (the building blocks of a transformer architecture) as indivisible units, overlooking possibilities of functional substructure learned within them. In this work, we introduce a more fine-grained perspective that decomposes these components into orthogonal singular directions, revealing superposed and independent computations within a single head or MLP. We validate our perspective on widely used standard tasks like Indirect Object Identification (IOI), Gender Pronoun (GP), and Greater Than (GT), showing that previously identified canonical functional heads, such as the "name mover," encode multiple overlapping subfunctions aligned with distinct singular directions. Nodes in a computational graph, that are previously identified as circuit elements show strong activation along specific low-rank directions, suggesting that meaningful computations reside in compact subspaces. While some directions remain challenging to interpret fully, our results highlight that transformer computations are more distributed, structured, and compositional than previously assumed. This perspective opens new avenues for fine-grained mechanistic interpretability and a deeper understanding of model internals.

1 Introduction

Language models (LMs) exhibit complex and often surprising capabilities across tasks [Radford et al., 2019, Brown et al., 2020, Li et al., 2023, Javaheripi et al., 2023, Touvron et al., 2023, Jiang et al., 2023, Grattafiori et al., 2024, Joshi et al., 2024], yet their internal computations remain poorly understood. Mechanistic interpretability seeks to bridge this gap by identifying the circuits, i.e., networks of interacting components, that realize/show specific functions [Olah et al., 2020, Wang et al., 2022]. Prior work has shown that these circuits can often be decomposed into submodules with distinct roles such as copying, inhibition, or referencing [Wang et al., 2022]. Despite this progress, the current state of circuit-discovery methods [Conmy et al., 2023, Syed et al., 2024, Bhaskar et al., 2024] still view model components, such as attention heads and MLPs, as atomic units of computation. Methods such as causal tracing [Meng et al., 2022], activation patching [Wang et al., 2022], and attribution-based analyses [Heimersheim and Nanda, 2024, Joshi et al., 2025b,a] typically probe/patch or ablate entire components to assess their functional contribution. While these techniques have produced valuable insights, they inherently assume that functionality aligns cleanly with component boundaries. In practice, however, transformer layers may multiplex multiple subfunctions within a single head or MLP, meaning that treating components as monolithic units risks overlooking the fine-grained structure of computation within these modules.

*Equal Contributions

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

While most existing methods analyze/study transformer components as monolithic units, recent work has begun to question this assumption by investigating the internal structure of these components. Merullo et al. [2024b], for instance, introduced a low-rank perspective showing that attention heads communicate through specific singular directions in residual space, defined by the singular vectors of their value matrices. However, this analysis primarily captures inter-component communication, how heads "talk" to one another via low-rank channels, while leaving intra-component decomposition largely unexplored, i.e., how a single head might multiplex multiple independent functions within its internal subspace.

In this work, we extend this perspective, which goes beyond the attention heads with additional inclusion of MLP layers, leading to a comprehensive directional view of transformer blocks. This reveals that low-rank, distributed computations are a general feature of transformer architectures. Moreover, components identified as part of known circuits [Wang et al., 2022, Conmy et al., 2023, Syed et al., 2024, Bhaskar et al., 2024] exhibit strong engagement along specific singular directions, suggesting that meaningful computations are embedded within compact subspaces. We further demonstrate/validate this perspective on the widely popular canonical tasks like Indirect Object Identification (IOI) [Wang et al., 2022], Gender Pronoun (GP) [Mathwin et al., 2023], and Greater Than (GT) [Hanna et al., 2023]. In IOI, for instance, our analysis identifies dominant singular directions within the same heads previously characterized as "name movers," [Wang et al., 2022], showing that only a sparse subset of these directions meaningfully contributes to task performance. Using our proposed optimization scheme, we learn direction-level masks that remain highly sparse while closely replicating the model's original behavior, indicating that transformer computations can be effectively captured by a compact set of low-rank subfunctions. Moreover, the directions corresponding to established IOI heads exhibit notably higher activation and mask weights compared to other heads, supporting the view that known circuit components operate through a small number of active, interpretable directions. In a nutshell, we make the following contributions:

- We introduce a directional interpretability perspective that models transformer components (attention and MLP) as superpositions of orthogonal subfunctions rather than atomic units.
- We demonstrate this via including an optimization-based masking scheme that identifies functionally important singular directions within attention and MLP layers, enabling direction-level attribution.
- We provide empirical evidence that multiple low-rank, interpretable computations coexist within single attention heads and MLPs, which diverge from the standard assumptions about circuit modularity.

Our findings suggest that transformer computations are not strictly modular but rather distributed, compact, and compositional, with overlapping subfunctions embedded within shared subspaces. This perspective reframes transformer interpretability through the lens of functional directions, opening a new avenue for analyzing, editing, and understanding model behavior at the subcomponent level.

Beyond component-level decomposition, our study/investigations reveal an interesting uncovering that transformer layers naturally form stable, controllable directions in logit space, each aligned with a specific set of tokens (which we also term as ***logit receptors***, more details in Appendix B.2). These directions can be thought of as intrinsic mechanisms that the model selectively activates depending on the input context. For example, in gender pronoun resolution, certain directions consistently influence the logits toward tokens "_he" or "_she," with their activation strengths varying systematically in response to context. Importantly, we find that just scalar interventions along these directions can reliably control/modify the model's predictions (also see Figure 2), demonstrating that these low-rank, interpretable subspaces may form interpretable building blocks of model computation. This insight provides a natural, mechanistic basis for studying how distributed computations within single heads and MLPs implement distinct functional behaviors, and motivates the fine-grained directional analysis presented in this work.

Overall, we believe that this perspective reframes how we think about transformer computations, i.e., rather than being confined to monolithic components, meaningful behaviors are often embedded in low-rank subspaces that can be independently manipulated and interpreted. This opens the door to more precise mechanistic studies, targeted model editing, and fine-grained attribution methods, and suggests that future interpretability work should consider the functional decomposition of components along intrinsic directions as a fundamental lens for understanding model behavior for abstract interpretable concepts learned by these models. We release our codebase for the experiments and additional results at `https://github.com/Exploration-Lab/Beyond-Components`.

2 A Unified Linear View of Transformer Components

To operationalize our directional interpretability perspective, we begin by expressing transformer computations in a unified linear form. We focus on decoder-only architectures and take the transformer circuit formulation from Elhage et al. [2021] as our foundation. By representing both attention and MLP transformations through augmented matrices that jointly include the learned weights and biases, we obtain a consistent linear representation across all components. This "folding in" of biases (i.e., appending a constant dimension to the input and incorporating bias terms into the matrix) allows us to apply Singular Value Decomposition (SVD) uniformly to both attention and MLP layers. The resulting formulation enables us to analyze orthogonal directions across different components within a shared framework, laying the groundwork for the fine-grained decomposition.

Attention Mechanism (Query Key (QK) Interaction) In a standard decoder-only transformer architecture [Radford et al., 2019], each attention head computes attention scores via the dot product between query and key row vectors

$$\alpha_{ij}^{(h)} = \text{Softmax}_j\left(\frac{\mathbf{q}_i^{(h)} \cdot \mathbf{k}_j^{(h)\top}}{\sqrt{d_{\text{head}}}}\right),$$

where $\mathbf{q}_i^{(h)} = \mathbf{x}_i \mathbf{W}_Q^{(h)} + \mathbf{b}_Q^{(h)}$ and $\mathbf{k}_j^{(h)} = \mathbf{x}_j \mathbf{W}_K^{(h)} + \mathbf{b}_K^{(h)}$. Expanding the dot product gives

$$\mathbf{q}_i\, \mathbf{k}_j^\top = \mathbf{x}_i \mathbf{W}_Q \mathbf{W}_K^\top \mathbf{x}_j^\top + \mathbf{x}_i \mathbf{W}_Q \mathbf{b}_K^\top + \mathbf{b}_Q \mathbf{W}_K^\top \mathbf{x}_j^\top + \mathbf{b}_Q \mathbf{b}_K^\top.$$

To express this interaction compactly as a single linear operation, we introduce an augmented matrix formulation that incorporates both weights and biases

$$[1, \mathbf{x}_i]\, \mathbf{W}_{\text{aug}}^{(QK)} \begin{bmatrix} 1 \\ \mathbf{x}_j^\top \end{bmatrix} = \mathbf{q}_i \cdot \mathbf{k}_j^\top.$$

where the augmented weight matrix is defined as

$$\mathbf{W}_{\text{aug}}^{(QK)} = \begin{pmatrix} \mathbf{b}_Q \mathbf{b}_K^\top & \mathbf{b}_Q \mathbf{W}_K^\top \\ \mathbf{W}_Q \mathbf{b}_K^\top & \mathbf{W}_Q \mathbf{W}_K^\top \end{pmatrix},$$

Attention Mechanism (Output Value (OV) Projection) Following the computation of attention weights, each head aggregates contextual information by taking a attention weighted sum of value vectors, and projecting it into the residual stream using $\mathbf{W}_O$. This operation determines how information is written back into the residual stream:

$$\mathbf{z}_i = \sum_j \alpha_{ij} \mathbf{v}_j, \qquad \text{where } \mathbf{v}_j = \mathbf{x}_j \mathbf{W}_V + \mathbf{b}_V.$$

Each head's contribution is then projected through its output matrix:

$$\mathbf{y}_i^{(h)} = \mathbf{z}_i \mathbf{W}_O^{(h)} + \frac{1}{|H|} \mathbf{b}_O.$$

where, $|H|$ is the number of heads. Substituting $\mathbf{v}_j$ and rearranging terms, the OV transformation becomes

$$\mathbf{y}_i^{(h)} = \sum_j \alpha_{ij} \left(\mathbf{x}_j \mathbf{W}_V \mathbf{W}_O^{(h)} + \mathbf{b}_V \mathbf{W}_O^{(h)}\right) + \frac{1}{|H|} \mathbf{b}_O = [1\ , \sum_j \alpha_{ij} \mathbf{x}_j] \begin{pmatrix} \mathbf{b}_V \mathbf{W}_O^{(h)} + \frac{1}{|H|} \mathbf{b}_O \\ \mathbf{W}_V \mathbf{W}_O^{(h)} \end{pmatrix} \tag{1}$$

We thus define the augmented output matrix for each head as

$$\mathbf{W}_{\text{aug}}^{(OV)} = \begin{pmatrix} \mathbf{b}_V \mathbf{W}_O^{(h)} + \frac{1}{|H|} \mathbf{b}_O \\ \mathbf{W}_V \mathbf{W}_O^{(h)} \end{pmatrix} \in \mathbb{R}^{(1+d_{\text{model}}) \times d_{\text{model}}}.$$

MLP Layer Reformulation Beyond attention, transformer MLP blocks also consist of two affine transformations separated by a nonlinearity. To maintain a consistent linear treatment across all components, we explicitly separate these two projections and represent each using augmented matrices that include both weights and biases:

$$\mathbf{y}_1 = \mathbf{x}\mathbf{W}_{\text{in}} + \mathbf{b}_{\text{in}}, \quad \mathbf{y}_{\text{out}} = f(\mathbf{y}_1)\mathbf{W}_{\text{out}} + \mathbf{b}_{\text{out}}.$$

The augmented representations are defined as:

$$\mathbf{W}_{\text{aug}}^{(\text{in})} = \begin{pmatrix} \mathbf{b}_{\text{in}} \\ \mathbf{W}_{\text{in}} \end{pmatrix}, \quad \mathbf{W}_{\text{aug}}^{(\text{out})} = \begin{pmatrix} \mathbf{b}_{\text{out}} \\ \mathbf{W}_{\text{out}} \end{pmatrix}.$$

Thus, both pre-activation and post-activation projections can be expressed as linear maps over augmented input vectors $[1, x]$, yielding a unified affine-to-linear transformation consistent with our treatment of attention layers.

This unified formulation enables all major transformer subcomponents to be represented as linear operators over augmented spaces, summarized as:

$$\begin{aligned}
\mathbf{W}_{\text{aug}}^{(QK)} &\in \mathbb{R}^{(1+d_{\text{model}})\times(1+d_{\text{model}})} && \text{: for attention score computation} \\
\mathbf{W}_{\text{aug}}^{(OV)} &\in \mathbb{R}^{(1+d_{\text{model}})\times d_{\text{model}}} && \text{: for weighted input-to-output transformation} \\
\mathbf{W}_{\text{aug}}^{(\text{in})} &\in \mathbb{R}^{(1+d_{\text{model}})\times d_{\text{mlp}}} && \text{: for MLP input projection} \\
\mathbf{W}_{\text{aug}}^{(\text{out})} &\in \mathbb{R}^{(1+d_{\text{mlp}})\times d_{\text{model}}} && \text{: for MLP output projection}
\end{aligned}$$

By expressing all these transformations in a common linear framework, we can perform low-rank analyses such as SVD across both attention and MLP components in a consistent manner. This perspective lays the groundwork for the directional interpretability approach introduced in the next section, where we analyze how specific singular directions correspond to distinct, functionally meaningful computations.

3 Directional Masking via Low-Rank Decomposition

Having established a unified linear formulation of transformer components (§2), we now turn to identifying the specific functional directions that drive model behavior. Our goal is to decompose each augmented attention or MLP matrix into a set of orthogonal directions, each representing an independent computational axis, and to selectively intervene on these axes to understand their roles.

Rather than treating attention heads or MLP layers as monolithic units, we perform decomposition on their augmented matrices, ensuring faithfulness to the model's native computation flow while enabling fine-grained directional attribution and masking. This formulation makes it possible to characterize a component's behavior not in terms of entire weight matrices, but in terms of a small number of low-rank directions.

Singular Value Decomposition (SVD): Any real matrix $M \in \mathbb{R}^{m\times n}$ admits a singular value decomposition $M = U\,\Sigma\,V^\top$, where $U \in \mathbb{R}^{m\times m}$ and $V \in \mathbb{R}^{n\times n}$ are orthogonal matrices, and $\Sigma \in \mathbb{R}^{m\times n}$ is a diagonal matrix of non-negative singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq 0$. This makes singular vectors a natural coordinate system for direction-level interpretability, as they provide orthonormal bases that isolate independent computational units embedded within a component.

Masking Directions: To isolate and investigate the contributions of individual computational directions, we apply SVD to the augmented matrices of each attention head and MLP layer:

$$\begin{aligned}
\mathbf{W}_{\text{aug}}^{(QK)} &= U_{QK}\,\Sigma_{QK}\,V_{QK}^\top, & \mathbf{W}_{\text{aug}}^{(OV)} &= U_{OV}\,\Sigma_{OV}\,V_{OV}^\top, \\
\mathbf{W}_{\text{aug}}^{(\text{in})} &= U_{\text{in}}\,\Sigma_{\text{in}}\,V_{\text{in}}^\top, & \mathbf{W}_{\text{aug}}^{(\text{out})} &= U_{\text{out}}\,\Sigma_{\text{out}}\,V_{\text{out}}^\top,
\end{aligned}$$

Each decomposition expresses the component as a sum of rank-1 mappings, where each term represents an orthogonal direction that can, in principle, support a distinct subfunction (also used as singular directions in the paper). To study the importance of these subfunctions, we introduce a learnable diagonal mask matrix $\mathcal{M} = \text{diag}(m_1, m_2, \cdots, m_r), m_i \in [0, 1]$, that scales the contribution of each singular direction. The masked transformation is then defined as:

$$\widetilde{\mathbf{W}}_{\text{aug}} = U\,\Sigma\,\mathcal{M}\,V^\top,$$

where higher entries of $\mathcal{M}$ retain or emphasize specific directions and lower entries suppress them. This allows continuous, direction-level control over component behavior while maintaining differentiability for optimization.

To ensure interpretability and stability, we preserve the full representational span of the component via:

$$U\,\Sigma\,V^\top = U\,\Sigma\,\mathcal{M}\,V^\top + U\,\Sigma\,(\mathcal{I}-\mathcal{M})\,V^\top$$

This decomposition allows faithful interventions, i.e., the model's downstream layers continue to receive inputs within their expected distribution (seen while training), since the overall covariance structure of activations remains intact. Thus, both masked and complementary subspaces are retained, enabling controlled investigation without inducing distributional drift.

$$\begin{aligned}
\widetilde{\mathbf{W}}^{(QK)}_{\text{aug}} &= U_{QK}\,\Sigma_{QK}\,\mathcal{M}_{QK}\,V_{QK}^\top && \in \mathbb{R}^{(1+d_{\text{model}})\times(1+d_{\text{model}})}\\
\widetilde{\mathbf{W}}^{(OV)}_{\text{aug}} &= \left[U_{OV}\,\Sigma_{OV}\,\mathcal{M}_{OV}\,V_{OV}^\top,\quad U_{OV}\,\Sigma_{OV}\,(\mathcal{I}-\mathcal{M}_{OV})\,V_{OV}^\top\right]^\top && \in \mathbb{R}^{(2(1+d_{\text{model}}))\times(d_{\text{model}})}\\
\widetilde{\mathbf{W}}^{(\text{in})}_{\text{aug}} &= \left[U_{\text{in}}\,\Sigma_{\text{in}}\,\mathcal{M}_{\text{in}}\,V_{\text{in}}^\top,\quad U_{\text{in}}\,\Sigma_{\text{in}}\,(\mathcal{I}-\mathcal{M}_{\text{in}})\,V_{\text{in}}^\top\right]^\top && \in \mathbb{R}^{(2(1+d_{\text{model}}))\times(d_{\text{mlp}})}\\
\widetilde{\mathbf{W}}^{(\text{out})}_{\text{aug}} &= \left[U_{\text{out}}\,\Sigma_{\text{out}}\,\mathcal{M}_{\text{out}}\,V_{\text{out}}^\top,\quad U_{\text{out}}\,\Sigma_{\text{out}}\,(\mathcal{I}-\mathcal{M}_{\text{out}})\,V_{\text{out}}^\top\right]^\top && \in \mathbb{R}^{(2(1+d_{\text{mlp}}))\times(d_{\text{model}})}
\end{aligned}$$

This procedure provides a principled mechanism for identifying the causally relevant subspaces within each transformer component, while keeping the model's computations structurally and statistically consistent with its pre-trained dynamics.

Note that for the QK matrices, we retain only the masked component $U_{QK}\,\Sigma_{QK}\,\mathcal{M}_{QK}\,V_{QK}^\top$ and omit the complementary term $U_{QK}\,\Sigma_{QK}\,(\mathcal{I}-\mathcal{M}_{QK})\,V_{QK}^\top$. This asymmetry arises from the distinct functional role of the QK block, whereas the OV and MLP matrices operate on feature representations, the QK matrices parameterize the attention kernel , the quadratic form that defines pairwise token similarities. Introducing a complementary $(\mathcal{I}-\mathcal{M})$ subspace here would correspond to defining additional, independent similarity maps within the same attention head. Because attention scores are normalized via a single softmax, this would implicitly produce multiple incompatible superimposed kernels, leading to interference and spurious correlations. In other words, if the true token representations $x=\{x_i,x_j\}$ become correlated with irrelevant or spurious features $x^{\text{corr}}=\{x_i^{\text{corr}},x_j^{\text{corr}}\}$, (which is often the case due to their similar framing/syntactic-style of x_i^{corr}), the mixed terms that arise from partially masked QK components can yield misleading similarity/attention scores: $s_{ij}\propto m\,x_iW_{QK}x_j^\top+(1-m)\,x_i^{\text{corr}}W_{QK}x_j^{\text{corr}\top}$. For instance, if $x_i^{\text{corr}}W_{QK}x_j^{\text{corr}\top}$ term produces a similarity score close to $x_iW_{QK}x_j^\top$, the training objective will incorrectly suppress the mask value ($m\to 0$) even if the component is genuinely useful for the task. Thus, we restrict QK masking to the primary $\mathcal{M}_{QK}$, which preserves a single, coherent attention kernel while still permitting direction-level ablations over its singular basis.

Optimization Objective. To identify the singular directions most responsible for a model's behavior on a given task, we optimize the learnable masks $\mathcal{M}$ to balance faithfulness (preserving model behavior) and sparsity (selecting only a minimal subset of directions).

Let $p(y\mid x)$ denote the original model's predictive distribution, and $p_{\mathcal{M}}(y\mid x)$ the distribution obtained after applying the masked matrices. We define the optimization objective as:

$$\mathcal{L}_{\mathcal{M}}=\text{KL}[p(y\mid x)\,\|\,p_{\mathcal{M}}(y\mid x)]+\lambda\,\|\text{diag}(\mathcal{M})\|_1,$$

where the KL divergence term encourages the masked model to reproduce the original model's behavior, and the ℓ_1-regularization term promotes sparsity in the mask, selecting only a minimal subset of singular directions. The trade-off coefficient λ governs the balance between behavioral consistency (faithfulness) and sparsity.

This procedure (summarized in Algorithm 1) yields a low-rank, sparse decomposition, where each retained singular direction captures a distinct, functionally meaningful axis of computation within the model's layers. While we use ℓ_1-based sparsity here, ℓ_0-regularization could provide an alternative sparse selection mechanism [Bhaskar et al., 2024, Sung et al., 2021], and is left for future work.

To improve robustness and better isolate task-relevant directions, we feed the model concatenated activations from both clean and corrupted inputs, where x denotes the clean activations from the original input, and x_{corrupt} denotes corresponding corrupted or perturbed activations (e.g., with noise or prompt that should consist of a small change to x that would result in a different label in the task). The combination x, x_{corrupt} forms a joint input, leading to the next token prediction $p_{\mathcal{M}}(y\mid x)=f\big(\sum_k(m_kx+(1-m_k)x_{corrupt})\sigma_ku_kv_k^\top\big)$ (details in Algorithm 1), enabling the mask optimization to identify singular directions that robustly capture task-relevant computations.

Algorithm 1 Directional Mask Optimization via Singular Value Decomposition

Require: Pretrained model f_θ, dataset $\mathcal{D} = \{(x, y)\}$, sparsity coefficient λ,

Initialize learnable diagonal masks $\mathcal{M}_{QK}, \mathcal{M}_{OV}, \mathcal{M}_{\text{in}}, \mathcal{M}_{\text{out}} \in [0,1]^{rank}$

for all components (attention heads and MLP layers) **do**

  Construct augmented weight matrix $\mathbf{W}_{\text{aug}}$

  Compute SVD: $\mathbf{W}_{\text{aug}} = U\,\Sigma\,V^\top$

end for

Freeze model parameters θ; keep masks $\mathcal{M}$ learnable

for each batch $(x, y) \in \mathcal{D}$ **do**

  Define helper function for joint representation:

$$\mathbf{h}(x, x_{\text{corrupt}}, \mathbf{W}) \;=\; \big[\,[1, x],\, [1, x_{\text{corrupt}}]\,\big]\,\mathbf{W}$$

  Define component-specific reconstruction functions, for each layer l:

$$\mathbf{g}^{(l)}_{\text{attn}} = \sum_{h=1}^{|heads|} \mathbf{h}\Big(\Big[\sum\nolimits_j \text{Softmax}_j\Big(\tfrac{[1,x_i]\widetilde{\mathbf{W}}^{QK,(l,h)}_{\text{aug}}[1,x_j]^\top}{\sqrt{d_{\text{head}}}}\Big)x_j\Big], x_{\text{corrupt}}, \widetilde{\mathbf{W}}^{OV,(l,h)}_{\text{aug}}\Big)$$

$$\mathbf{g}^{(l)}_{\text{mlp}} = \mathbf{h}\Big(\text{GeLU}\big(\mathbf{h}(x, x^{in}_{\text{corrupt}}, \widetilde{\mathbf{W}}^{\text{in},(l)}_{\text{aug}})\big),\, x^{out}_{\text{corrupt}},\, \widetilde{\mathbf{W}}^{\text{out},(l)}_{\text{aug}}\Big)$$

  Run forward pass through masked model using both contributions:

$$p_{\mathcal{M}}(y \mid x) = \text{Softmax}\left(\text{LayerNorm}\left[\text{embedding} + \sum_l \left(\mathbf{g}^{(l)}_{\text{attn}} + \mathbf{g}^{(l)}_{\text{mlp}}\right)\right] W_U + b_U\right)$$

  Compute objective:

$$\mathcal{L}_{\mathcal{M}} = \text{KL}[p(y \mid x) \,\|\, p_{\mathcal{M}}(y \mid x)] + \lambda \,\|\text{diag}(\mathcal{M})\|_1$$

  Update $\mathcal{M}$ via gradient descent, and and reconstruct weights $\widetilde{\mathbf{W}}_{\text{aug}}$

end for

return Learned masks $\mathcal{M}$ and reconstructed weights $\widetilde{\mathbf{W}}_{\text{aug}}$

Note: x (row vector) denotes the running forward pass activations while optimization, and x_{corrupt} (row vector) denotes the appropriate corresponding corrupted or perturbed activations, that are fixed and obtained from a corrupted run.

4 Experiments

Having introduced directional masking, which decomposes each transformer component into orthogonal singular directions and enables selective interventions, we now ask: *do these fine-grained directions correspond to meaningful, task-relevant subfunctions within the model?*

Prior interpretability studies have associated specific attention heads or MLP layers with distinct behaviors, such as syntactic role tracking or inhibition [Wang et al., 2022, Conmy et al., 2023]. However, these analyses treat components as atomic units. In contrast, our approach allows us to examine computation at the level of individual singular directions, revealing whether components multiplex multiple independent computations along distinct low-rank axes.

We apply our method to a pretrained GPT-2 Small model [Radford et al., 2019], a tractable benchmark widely used in mechanistic interpretability [Wang et al., 2022, Hanna et al., 2023]. To inspect the generality of directional subfunctions, we evaluate the model on three representative tasks: Indirect Object Identification (IOI) [Wang et al., 2022], which tests syntactic reasoning by examining coreference resolution in sentences. Greater-Than (GT) [Hanna et al., 2023], where the model predicts numerical tokens following a number in context, assessing its quantitative reasoning. Gender Pronoun Resolution (GP) [Mathwin et al., 2023], which measures semantic reasoning by testing pronoun-to-antecedent resolution in natural text. Full dataset details are provided in App. A.

We organize our experiments around four core research questions. **R1) Can** a small number of learned singular directions faithfully preserve model behavior? **R2) Do** these directions align with

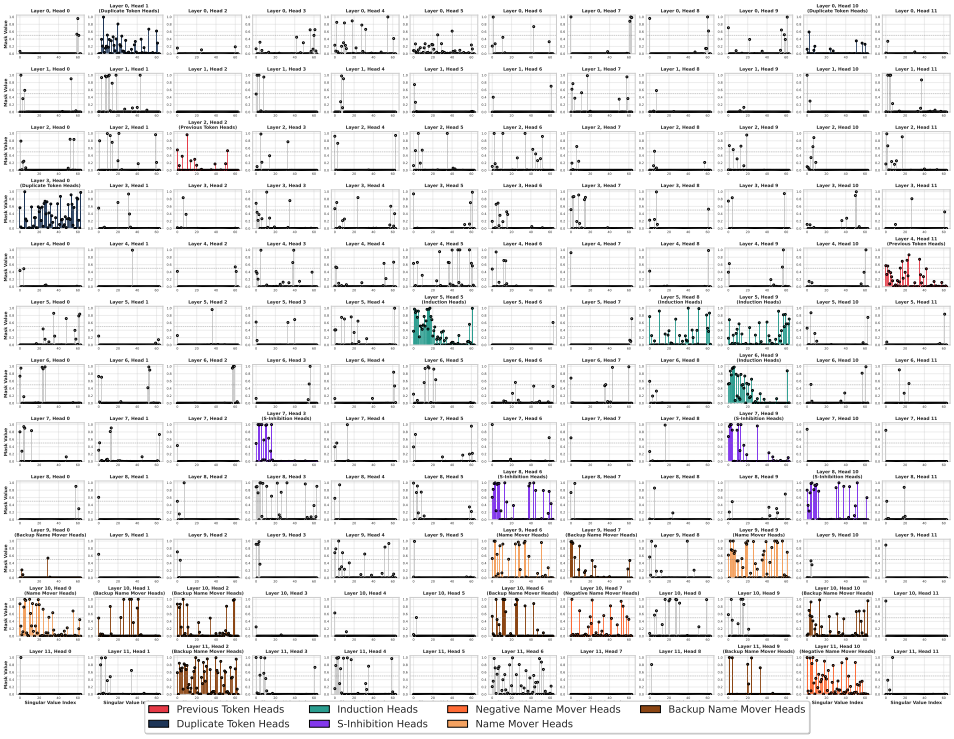

Figure 1: Learned singular value masks for Query-Key ($\mathbf{W}_{\mathbf{aug}}^{\mathbf{QK}}$) matrices across all attention heads in the model. High mask activations correspond to circuit components previously identified for the IOI task [Wang et al., 2022]. Each head exhibits sparsity along its singular directions, revealing the fine-grained subspaces driving task behavior.

components identified in prior circuit-level analyses? **R3) Can** our method decompose known functional heads into interpretable sub-functions? **R4) Can** we discover new functional axes not identifiable through standard component-level analysis?

For each attention and MLP component in the GPT-2 Small model, we extract its augmented matrices $\mathbf{W}_{\text{aug}}^{(QK)}$, $\mathbf{W}_{\text{aug}}^{(OV)}$, $\mathbf{W}_{\text{aug}}^{(in)}$ and, $\mathbf{W}_{\text{aug}}^{(out)}$ (as described in §2) and perform a singular value decomposition. We retain all non-zero singular directions, corresponding to the actual matrix rank rather than a fixed truncation. For the Query–Key ($\mathbf{W}_{\text{aug}}^{QK}$) matrices, the effective rank is 64, hence $r = 64$ reflects the full set of functional directions. Similarly, the Output-Value ($\mathbf{W}_{\text{aug}}^{OV}$) matrices exhibit a rank of 65 after augmentation, while MLP layers typically retain their full non-zero spectrum. Empirically, we found these truncations yield minuscule drops (in the range of $(1e-6)$) in reconstruction faithfulness (KLD), confirming that the effective ranks of the augmented matrices capture all functionally relevant subspaces.

Over these singular directions, we introduce a learnable diagonal mask that scales the singular values, enabling direction-level modulation of each component's contribution. The masks are optimized using the objective defined in §3. The optimization in Algorithm 1 proceeds on mini-batches drawn from each dataset, and optimization is stopped early once the held-out reconstruction loss stabilizes. For corruption (x_{corrupt}), we follow the datasets provided by Bhaskar et al. [2024], and create an additional set of datapoints for tasks GT and GP.

Our first experiment evaluates whether the model's behavior can be faithfully reconstructed using only a small subset of learned singular directions (Table 1). Remarkably, across all datasets, high-fidelity reconstruction is possible with far fewer directions than the full component. For instance, the IOI task retains only $\sim$ 9% of directions relative to the full component while achieving a KL divergence of 0.21 and an exact match of 0.77, demonstrating that a small fraction of singular directions suffices to reproduce the model's behavior. Similar patterns hold for GT and GP, where over 95% of directions can be pruned with minimal impact on performance. The learned masks are distinctly sparse, activating only a handful of directions per layer.

Table 1: Comparison of sparsity, reconstruction fidelity, and task performance across datasets. Sparsity is measured relative to the number of non-zero singular directions in each component ("Relative") and as a fraction of the full matrix size ("Full"). KL divergence quantifies reconstruction loss, while accuracy (where applicable) and exact match show downstream task performance. Despite extreme sparsity, the learned directions retain high behavioral fidelity. (see App. B.6 for sparsity computations).

Dataset	Sparsity (Rel / Full)	KLD	Acc. (Pruned / Full)	Exact Match
IOI	91.32 / 98.66	0.21±0.02	0.70±0.07 / 0.79±0.05	0.77±0.06
GT	95.21 / 99.26	0.23±0.03	N/A	0.33±0.06
GP	96.81 / 99.51	0.13±0.01	0.75±0.04 / 0.77±0.04	0.86±0.07

These selected directions achieve substantially lower KL divergence than top-k magnitude or random SVD baselines, suggesting that task-relevant computation is concentrated along specific, semantically meaningful directions rather than the largest singular modes (quantitative details in App. B).

Next, we investigate how our learned singular directions correspond to previously identified mechanistic circuits, using the IOI task as a reference and comparing to ACDC [Conmy et al., 2023] and Wang et al. [2022]. While prior analyses treat entire attention heads as circuit participants, our direction-level decomposition reveals that most heads exhibit strong activation along only a few singular directions. In other words, the coarse, component-level circuits reported in previous work arise from finer, low-rank structures embedded within each head.

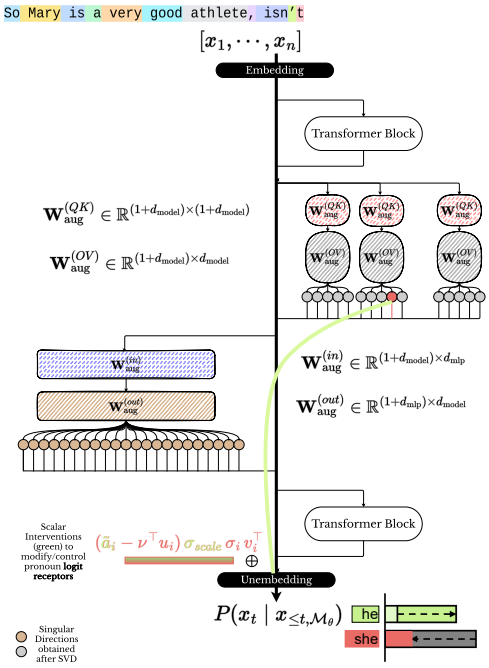

Figure 2: The Figure shows intervention in the logit receptor for the Gender Pronoun task. Controlling the logit receptor using a scalar intervention modifies the predicted logits.

Figures 1, 5, and 6 visualize the learned singular-value masks for Query-Key, OV, and MLP matrices, respectively, across the full model. Heads known to play key roles in IOI, such as Name Mover, Backup Name Mover, and S-Inhibition Wang et al. [2022], show consistently high activations across multiple singular directions, pinpointing the precise subspaces driving their behavior. Conversely, components not associated with known circuits exhibit near-zero activations across all directions (App. Figure 7), demonstrating that our optimization selectively isolates task-relevant subspaces.

Overall, these results demonstrate that transformer components are not monolithic, and their internal computation is distributed along a small number of interpretable, low-rank axes/bases. The presented perspective thus bridges component-level circuit discovery and fine-grained mechanistic analysis, providing a new lens to study how distinct computational roles coexist within the same architectural unit.

Extending this perspective, we find that transformer layers contain inherent fixed directions in logit space corresponding to stable token-preference axes. Task behavior emerges from dynamically steering these directions via input-dependent scalar activations; in the Gender Pronoun task, for instance, distinct "he" and "she" directions exist, and the model selectively activates them depending on context. Scalar-based interventions confirm that these directions are causally relevant by flipping gender pronoun predictions with perfect accuracy, demonstrating that low-rank subspaces serve as modular, functional building blocks bridging representation and output behavior. All these results provide a unified mechanistic view, i.e., computation is concentrated along sparse, interpretable

singular directions within components, which in turn map to stable, task-relevant axes in logit space (full details in App. B.2).

5 Functional Decomposition of Attention Head 9.6 via Singular Directions

Building on our discovery that transformer components distribute computation along low-rank, interpretable directions, we next ask: what specific operations does a single head implement through these directions? We focus on Head 9.6, previously identified as a key contributor to the IOI task [Wang et al., 2022, Conmy et al., 2023, Bhaskar et al., 2024], and examine its singular directions with high learned mask values. Our analysis reveals individual singular directions encode distinct, separable operations, ranging from syntactic cues to semantic entity tracking, that collectively support task performance. Table 9 summarizes the primary roles and their corresponding mask values, highlighting how these directions act as interpretable computational primitives within a single attention head.

Semantic Separation of Entities and Actions Some singular directions carve out conceptually meaningful subspaces, revealing that attention heads implement abstract computations rather than just capturing statistical variance. A striking example is the 7^{th} singular direction (S_7) in Head 9.6, which consistently separates named entities (e.g., "Mary", "Kevin") from action-related tokens (e.g., "went", "gave"). As shown in Table 7, entity tokens exhibit strong positive activations ($+3.52 \pm 1.42$), while action tokens are suppressed (-4.44 ± 0.68). For instance, in the sentence "Jerry and Mary went to the school. Mary gave a raspberry to", S_7 assigns high activations to "Jerry" ($+2.87$) and "Mary" ($+2.78$), but negative values to "went" (-4.33) and "gave" (-4.17). This direction acts like a semantic filter, splitting "who" from "what is being done," creating a foundation upon which downstream heads can operate.

Entity Salience and Detection Another key computation is performed by the 28^{th} singular direction (S_{28}), which detects and amplifies the salience of named entities. In IOI prompts, tracking participants across multiple mentions is critical, and S_{28} highlights entities regardless of their position or specific lexical form. Table 8 shows that named entities receive high activations (e.g., "Susan": 4.05, "Kevin": 5.22), whereas function words like "the" and "of" remain low. Interestingly, first mentions often get stronger activations than subsequent mentions (e.g., "Kevin": 5.22, "Kevin_2": 2.07), suggesting this direction is sensitive to positional salience. We interpret S_{28} as an entity salience signal, priming tokens for further grammatical and referential reasoning.

Sequence Initialization Detection The top singular direction (S_1) implements a structural, positional primitive, i.e., it assigns extremely high positive values to the first token in a sequence while giving negligible or negative values to all others. Across prompts, the first token receives activations 20–25$\times$ larger than subsequent tokens. This behavior is not unique to Head 9.6; similar patterns appear in other heads and layers, independent of task, e.g., S_1 of Head 0 in Layer 10, suggesting that sequence initialization is a reusable primitive distributed across the network.

Summary and Implications All these examples show that Head 9.6 multiplexes multiple independent computations through orthogonal singular directions: semantic discrimination (S_7), entity salience (S_{28}), and sequence initialization (S_1). This supports the hypothesis that transformer components are integration points of overlapping, low-rank subfunctions, rather than monolithic units. The directional perspective provides a fine-grained lens for mechanistic interpretability, isolating the entangled computations within attention heads that would otherwise appear inseparable. Figure 8 summarizes the functional roles identified. Additional findings in App. B reinforce the idea that circuits should be understood not at the level of whole heads, but as a collection of directions, each contributing in distinct proportions to task-relevant behavior.

6 Related Work

Mechanistic interpretability seeks to break down neural networks into human-understandable components. Prior work has uncovered specialized attention heads and MLP circuits supporting tasks like indirect object identification and fact recall, typically treating heads or layers as atomic units [Elhage et al., 2021, Wang et al., 2022, Meng et al., 2022]. Our work complements this by going inside each head using singular value decomposition (SVD) on augmented weight matrices. This reveals independent computational directions, each implementing a distinct function, such as semantic

separation, entity salience, or sequence initialization in head 9.6. While past SVD-based analyses focused on isolated components [Gao et al., 2024, Cunningham et al., 2023], we generalize the approach across all core circuits, query-key, output-value, and MLP transformations, showing that individual directions, rather than entire heads, can carry task-relevant computations. This perspective also complements studies showing interdependencies across heads [Merullo et al., 2024a]. Instead of modeling communication between heads, we decompose each head internally, highlighting low-rank primitives that drive behavior. In doing so, we introduce a direction-level granularity, which provides a finer, mechanistically meaningful lens on model computation and opens a path toward understanding how overlapping subfunctions combine within a single attention head.

7 Discussion and Limitations

Rethinking Transformer Components. Our work emphasizes a simple but powerful idea: transformer components are not monolithic units, but rather collections of independent functional directions. Each direction can encode a distinct computational primitive, such as sequence initialization, entity salience, or semantic separation, allowing a single head or MLP block to multiplex multiple sub-functions. By zooming in on these low-rank directions, we can reveal hidden structure that is invisible when treating heads or layers as atomic, providing a finer-grained understanding of how transformers compute. This perspective has broad implications. It suggests that mechanistic interpretability should move beyond unit-level analysis, toward frameworks that capture both the individual roles of directions and their interactions, potentially defining new "micro-circuits" within components. Future work could explore direction-level communication patterns, drawing inspiration from recent studies on head-to-head interaction graphs [Merullo et al., 2024b], to understand how these sub-functions combine into higher-level behaviors.

Limitations: The directional decomposition perspective we present provides a fine-grained view into transformer computations; however, several limitations remain for further exploration. First, our analysis considers each augmented matrix independently, focusing on individual layers and heads. This isolation facilitates precise attribution of function to specific singular directions but may obscure emergent behaviors that arise from interactions across components, which are principal to many transformer computations. Second, the approach assumes that singular directions correspond to interpretable and causally relevant subroutines. Although supported by observed semantic boundaries and activation patterns, this assumption lacks formal justification, and task-relevant computation may be distributed across weaker directions or higher-order interactions. Third, our method applies learnable diagonal masks to fixed singular vectors, thereby restricting optimization to axis-aligned subspaces, which may limit expressiveness. Allowing controlled perturbations of singular directions could capture finer task-dependent variations. Fourth, our evaluation is limited to the standard benchmark tasks and GPT-2 Small. Whether the method scales to more complex reasoning tasks, larger models, or instruction-tuned systems remains an open question. Finally, while our method reveals functionally interpretable directions, it does not yet fully elucidate the causal role of each direction in the end-to-end model behavior. Our masking objective encourages output faithfulness, but disentangling true causal mediation from representational correlation remains an open challenge. Future work could integrate causal tracing or activation patching to more directly establish mechanistic influence.

Despite these limitations, directional decomposition provides a new lens on transformer internals. By treating model components as collections of functional directions, we can uncover the computational primitives that underlie complex behaviors, laying the foundation for more systematic and fine-grained circuit discovery in modern language models.

8 Conclusion

Transformers are often seen as black boxes, complex computational graphs of neurons, heads, and layers whose inner workings are difficult to disentangle. In this work, we demonstrate that a simple shift in perspective, focusing on individual singular directions in parameter space, can reveal surprisingly clean and interpretable sub-computations. This perspective also suggests a broader narrative. Might we someday construct compact, modular explanations of entire model behaviors by stitching together just a few interpretable directions? If so, we may one day piece together modular explanations of entire model behaviors from a small number of interpretable directions.

Acknowledgments

We would like to thank the anonymous reviewers and the meta-reviewer for their insightful comments and suggestions. This research work was partially supported by the Research-I Foundation of the Department of CSE at IIT Kanpur.

References

Adithya Bhaskar, Alexander Wettig, Dan Friedman, and Danqi Chen. Finding transformer circuits with edge pruning. In The Thirty-eighth Annual Conference on Neural Information Processing Systems, 2024. URL https://openreview.net/forum?id=8oSY3rA9jY. [Cited on pages 1, 2, 5, 7, and 9.]

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. CoRR, abs/2005.14165, 2020. URL https://arxiv.org/abs/2005.14165. [Cited on page 1.]

Arthur Conmy, Augustine N. Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. Towards automated circuit discovery for mechanistic interpretability. In Thirty-seventh Conference on Neural Information Processing Systems, 2023. URL https://openreview.net/forum?id=89ia77nZ8u. [Cited on pages 1, 2, 6, 8, and 9.]

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models, 2023. URL https://arxiv.org/abs/2309.08600. [Cited on page 10.]

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. https://transformer-circuits.pub/2021/framework/index.html. [Cited on pages 3 and 9.]

Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders, 2024. URL https://arxiv.org/abs/2406.04093. [Cited on page 10.]

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas

Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay,

Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783. [Cited on page 1.]

Michael Hanna, Ollie Liu, and Alexandre Variengien. How does gpt-2 compute greater-than?: Interpreting mathematical abilities in a pre-trained language model. Advances in Neural Information Processing Systems, 36:76033–76060, 2023. [Cited on pages 2, 6, 19, and 29.]

Michael Hanna, Sandro Pezzelle, and Yonatan Belinkov. Have faith in faithfulness: Going beyond circuit overlap when finding model mechanisms. In First Conference on Language Modeling, 2024. URL https://openreview.net/forum?id=TZ0CCGDcuT. [Cited on page 33.]

Stefan Heimersheim and Neel Nanda. How to use and interpret activation patching, 2024. URL https://arxiv.org/abs/2404.15255. [Cited on page 1.]

Aliyah R. Hsu, Georgia Zhou, Yeshwanth Cherapanamjeri, Yaxuan Huang, Anobel Odisho, Peter R. Carroll, and Bin Yu. Efficient automated circuit discovery in transformers using contextual decomposition. In The Thirteenth International Conference on Learning Representations, 2025. URL https://openreview.net/forum?id=41HlN8XYM5. [Cited on page 33.]

Mojan Javaheripi, Sébastien Bubeck, Marah Abdin, Jyoti Aneja, Caio César Teodoro Mendes, Weizhu Chen, Allie Del Giorno, Ronen Eldan, Sivakanth Gopi, Suriya Gunasekar, Piero Kauffmann, Yin Tat Lee, Yuanzhi Li, Anh Nguyen, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Michael Santacroce, Harkirat Singh Behl, Adam Taumann Kalai, Xin Wang, Rachel Ward, Philipp Witte, Cyril Zhang, and Yi Zhang. Phi-2: The surprising power of small language models. Microsoft Research Blog, 2023. [Cited on page 1.]

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, 2023. URL https://arxiv.org/abs/2310.06825. [Cited on page 1.]

Abhinav Joshi, Areeb Ahmad, and Ashutosh Modi. COLD: Causal reasoning in closed daily activities. In The Thirty-eighth Annual Conference on Neural Information Processing Systems, 2024. URL https://openreview.net/forum?id=7Mo1NOosNT. [Cited on page 1.]

Abhinav Joshi, Areeb Ahmad, and Ashutosh Modi. Calibration across layers: Understanding calibration evolution in LLMs. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 14697–14725, Suzhou, China, November 2025a. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.742. URL https://aclanthology.org/2025.emnlp-main.742/. [Cited on page 1.]

Abhinav Joshi, Areeb Ahmad, Divyaksh Shukla, and Ashutosh Modi. Towards quantifying commonsense reasoning with mechanistic insights. In Luis Chiruzzo, Alan Ritter, and Lu Wang, editors, Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 9633–9660, Albuquerque, New Mexico, April 2025b. Association for Computational Linguistics. ISBN 979-8-89176-189-6. doi: 10.18653/v1/2025.naacl-long.487. URL https://aclanthology.org/2025.naacl-long.487/. [Cited on page 1.]

Yuanzhi Li, Sébastien Bubeck, Ronen Eldan, Allie Del Giorno, Suriya Gunasekar, and Yin Tat Lee. Textbooks are all you need ii: phi-1.5 technical report, 2023. URL https://arxiv.org/abs/2309.05463. [Cited on page 1.]

Chris Mathwin, Guillaume Corlouer, Esben Kran, Fazl Barez, and Neel Nanda. Identifying a preliminary circuit for predicting gendered pronouns in gpt-2 small. URL: https://itch.io/jam/mechint/rate/1889871, 2023. [Cited on pages 2, 6, and 19.]

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. Advances in Neural Information Processing Systems, 35, 2022. [Cited on pages 1 and 9.]

Jack Merullo, Carsten Eickhoff, and Ellie Pavlick. Circuit component reuse across tasks in transformer language models, 2024a. URL https://arxiv.org/abs/2310.08744. [Cited on page 10.]

Jack Merullo, Carsten Eickhoff, and Ellie Pavlick. Talking heads: Understanding inter-layer communication in transformer language models. In The Thirty-eighth Annual Conference on Neural Information Processing Systems, 2024b. URL https://openreview.net/forum?id=LUsx0chTsL. [Cited on pages 2 and 10.]

Neel Nanda and Joseph Bloom. Transformerlens. https://github.com/TransformerLensOrg/TransformerLens, 2022. [Cited on page 20.]

Neel Nanda, Chris Olah, Catherine Olsson, Nelson Elhage, and Hume Tristan. Attribution patching: Activation patching at industrial scale, 2023. URL https://www.neelnanda.io/mechanistic-interpretability/attribution-patching. [Cited on page 33.]

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. https://distill.pub/2020/circuits/zoom-in. [Cited on page 1.]

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Estelle Tompson, Laurens Desmaison, Alban Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilimbi, and Benoit Prabhakaran. Pytorch: An imperative style, high-performance deep learning library. In Advances in neural information processing systems, pages 8024–8035, 2019. [Cited on page 19.]

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners, 2019. URL https://api.semanticscholar.org/CorpusID:160025533. [Cited on pages 1, 3, and 6.]

Yi-Lin Sung, Varun Nair, and Colin Raffel. Training neural networks with fixed sparse masks, 2021. URL https://arxiv.org/abs/2111.09839. [Cited on page 5.]

Aaquib Syed, Can Rager, and Arthur Conmy. Attribution patching outperforms automated circuit discovery. In Yonatan Belinkov, Najoung Kim, Jaap Jumelet, Hosein Mohebbi, Aaron Mueller, and Hanjie Chen, editors, Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP, pages 407–416, Miami, Florida, US, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.blackboxnlp-1.25. URL https://aclanthology.org/2024.blackboxnlp-1.25/. [Cited on pages 1 and 2.]

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023. URL https://arxiv.org/abs/2307.09288. [Cited on page 1.]

Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: a circuit for indirect object identification in gpt-2 small, 2022. URL https://arxiv.org/abs/2211.00593. [Cited on pages 1, 2, 6, 7, 8, 9, 17, 18, 19, 25, 26, 27, and 29.]

Appendix

Table of Contents

List of Tables

List of Figures

A Details of Datasets

This section provides details about the datasets used in our experiments. While the Indirect Object Identification (IOI) task is our primary benchmark due to its strong alignment with our goals in mechanistic interpretability, we also include two additional datasets, Greater-Than [Hanna et al., 2023] and Gender Pronoun [Mathwin et al., 2023], to test the generalizability of our method.

A.1 Indirect Object Identification (IOI)

The IOI task [Wang et al., 2022] is a synthetic benchmark designed to study a language model's ability to resolve coreference between proper names in complex syntactic constructions. Each prompt contains two names in an introductory clause, followed by a clause in which one name acts as the subject. The model must complete the sentence with the indirect object:

> *"When Mary and John went to the store, John gave a drink to"* $\rightarrow$ **Mary**

The task follows a known and interpretable algorithm, i.e., predict the name that is **not** the subject of the last clause. Its controlled structure and clear ground truth make it ideal for circuit-level analysis and interpreting internal representations of models like GPT-2.

A.2 Gender Pronoun Task

This task evaluates pronoun resolution in socially relevant contexts. Each example is a declarative sentence about a named person followed by a tag question. The goal is to complete the sentence with the correct gendered pronoun:

> *"So David is a really great friend, isn't"* $\rightarrow$ **he**
> *"So Mary is a very good athlete, isn't"* $\rightarrow$ **she**

This dataset complements IOI by testing the model's ability to resolve pronouns and capture gender semantics. It also provides insight into whether our method can disentangle social biases embedded in model representations.

A.3 Greater-Than Task

The Greater-Than task [Hanna et al., 2023] targets numerical reasoning by prompting the model with a sentence involving two years. The goal is to predict a valid end year that is greater than the start year, using a textual format:

> *"The treaty lasted from the year 1314 to the year 13"* $\rightarrow$ **28**

The dataset is automatically generated using 120 nouns representing temporal events (sourced from FrameNet), with years sampled from the 11th to 17th centuries. Special care is taken to avoid multi-token year completions and boundary cases, ensuring all completions are single-token and meaningful under GPT-2's tokenizing scheme.

B Additional Results, Discussion and Future Directions

In this appendix section, we provide extended results, analysis, and exploratory observations that complement the findings in the main paper. We provide further support for the interpretability of low-rank directions across multiple tasks, highlight composite functional patterns, and outline promising directions for future work.

B.1 Compute Resources

All experiments were conducted on a single NVIDIA A40 GPU with 48 GB of VRAM. Our implementation is based on PyTorch 2.1 [Paszke et al., 2019] and the HuggingFace Transformers library (v4.35), leveraging its integration with pre-trained models and tokenizer utilities. We utilize the GPT-2 small model (124M parameters), with all weights frozen during our experiments to ensure

Table 2: Dataset splits for the three tasks used in our experiments, IOI (Indirect Object Identification), GT (Gender Type), and GP (Gender Pronouns), indicating the number of examples allocated to the training, validation, and test sets.

Task	Train	Validation	Test
IOI	1k	200	1k
GT	2k	500	2k
GP	1k	155	307

Table 3: Hyperparameters used for training the linear probes and other learned components in our experiments. We report batch size, number of epochs, optimization parameters (learning rate and weight decay), and the coefficient of the L1 regularization term.

Hyperparameter	Value
Batch Size	64
Number of Epochs	15
Learning Rate	1.0×10^{-2}
Weight Decay	1.0×10^{-9}
L1 Regularization Weight	1.5×10^{-4}

the integrity of the underlying representations. For systematic access to internal model activations and component-wise analysis, we employ the `TransformerLens` library [Nanda and Bloom, 2022], which provides fine-grained control over the transformer's intermediate computations, enabling singular value decomposition and mask-based interventions at the component level. All training for our mask optimization and probing modules was conducted using batches of IOI or Greater-Than task prompts, with early stopping based on validation loss to ensure efficiency and generalizability. The details to the Dataset splits and Hyperparameters are provided in Table 2 and Table 3, respectively. Table 2 outlines the dataset splits used across different tasks, Indirect Object Identification (IOI), Greater-Than (GT), and Generalized Preposition (GP). We use relatively small training sets to textitasize the interpretability and efficiency of our method, while validation and test sets are significantly larger to ensure robust generalization. The training procedure for our low-rank component masks and probing modules was consistent across tasks, with hyperparameters detailed in Table 3. We optimize using AdamW with moderate weight decay and L1 regularization to encourage sparsity in learned masks. Training is conducted for a maximum of 150 epochs with early stopping based on validation performance to avoid overfitting. The relatively small batch size and learning rate were empirically found to balance convergence speed with stability.

B.2 Inherently formed Logit Directions

Beyond component-level decomposition, our analysis reveals that in transformer layers there exist a set of *inherent, fixed directions* in logit space, which we term as *"Logit Receptors"* (so named because they act as fixed receptive directions that deterministically modulate specific token logits). These are directions that correspond to stable vocabulary preferences learned during training and that operate as modulators for specific tokens. These directions emerge naturally from the singular value decomposition of the augmented OV projection, $\mathbf{W}_{\text{aug}}^{(\text{OV})}$, which writes the attention output into the residual stream.

From Singular Vectors to Logit Receptors. Let $\nu_{lh} \in \mathbb{R}^{(1+d_{\text{model}})}$ denote the attention-weighted (and one-augmented, to compensate bias) value vector for layer l, head h, where we augment the context with a constant 1 so that $\nu_{lh}^{\top} = [1, \sum_j \alpha_{ij} x_j]$ in the notation above(α is attention distribution and x_j is post-layernorm residual stream representation). Decomposing the corresponding OV projection by singular value decomposition gives

$$\mathbf{W}_{\text{aug}}^{(\text{OV})} = U_{lh}\, \Sigma_{lh}\, V_{lh}^{\top}.$$

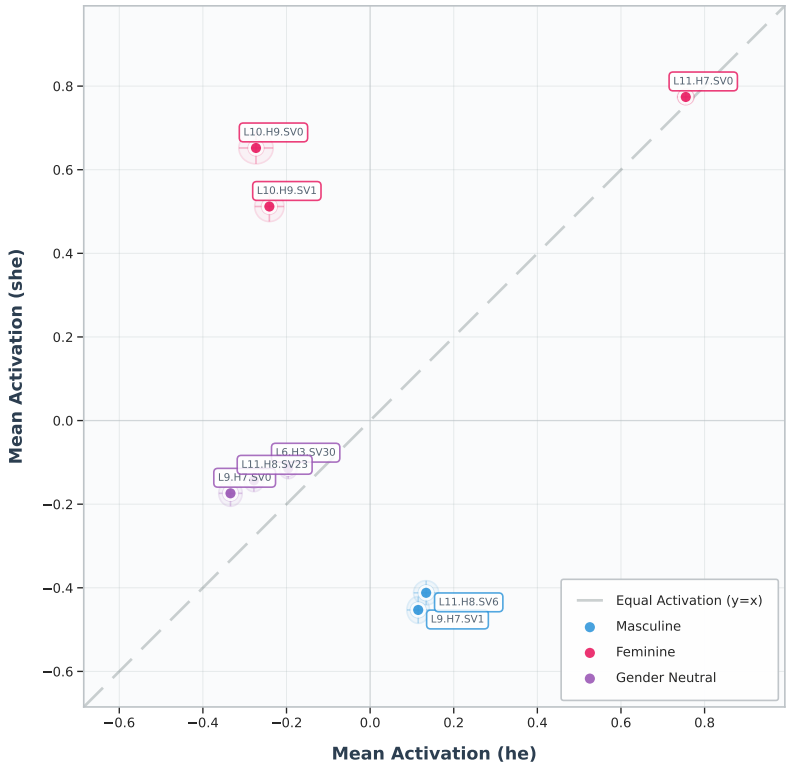

Figure 3: Mean activation of gender-related directions conditioned on *Masculine* versus *Feminine* prompt context. The x-axis plots the mean activation $\mathbb{E}[\nu^\top u \mid \text{prompt context=he}]$ and the y-axis plots mean $\mathbb{E}[\nu^\top u \mid \text{prompt context=she}]$. Error bars show one standard deviation. The dashed diagonal line represents $y = x$, where activations for both pronouns would be equal.

The output, in equation (1), written by head h in layer l in the residual stream can then be expressed as

$$y_{lh} = \nu_{lh}^\top \mathbf{W}_{\text{aug}}^{(\text{OV})} = \sum_k \sigma_{lhk} \left(\nu_{lh}^\top u_{lhk}\right) v_{lhk}^\top,$$

where each term corresponds to one singular direction k with singular value σ_{lhk}, left singular vector u_{lhk}, and right singular vector v_{lhk}. The scalar term $\nu_{lh}^\top u_{lhk}$ represents an input-dependent coefficient that steers a fixed direction in residual space.

When each right-singular vector v_{lhk} is projected through the unembedding matrix W_U, we obtain a scaled vocabulary-length vector

$$y_{lhk} = \left(\sigma_{lhk} \left(\nu_{lh}^\top u_{lhk}\right)\right) v_{lhk}^\top W_U,$$

which corresponds to a fixed direction (pointing towards a specific set of tokens) in token-logit space. Importantly, the term $v_{lhk}^\top W_U$ depends only on the learned weights; in other words, it is a constant Logit Receptor independent of any specific input. The only input-dependent term is its scalar activation, $\nu_{lh}^\top u_{lhk}$, which modulates how strongly the model calls upon that logit receptor at a given step.

Interpretable Logit Receptors. Empirically, we found that some of these fixed logit receptors have clear linguistic meaning. In the Gender Pronoun Resolution (GP) task, for instance, certain $v_{lhk}^\top W_U$ vectors are more inclined towards gendered pronouns. Their corresponding activations $\nu_{lh}^\top u_{lhk}$ are systematically higher for examples of one gender than the other, forming a mechanistic pattern, i.e., the model maintains distinct "she" and "he" receptors in logit space and routes activation selectively between them depending on context.

This perspective reveals a fresh way of understanding concepts present inherently in the internal structure of the model, even before considering the mask-learning step; the network already contains

Table 4: Selected gender- and number-related OV singular directions. Columns show the learned mask, singular value σ, an excerpt of top tokens from $v^\top W_U$, mean activation $\nu^\top u$ (mean $\pm$ std) conditioned on the pronoun based on prompt context(p_c), and the activation difference (he minus she).

Direction	Mask	σ	Top tokens (excerpt)	$\mathbb{E}[\nu^\top u \mid p_c$=he]	$\mathbb{E}[\nu^\top u \mid p_c$=she]	Diff
`L9.H7.SV1`	1.00	8.87	His, his, He, he, himself	$+0.115 \pm 0.027$	-0.453 ± 0.032	$+0.568$
`L11.H8.SV6`	1.00	6.52	his, His, him, He, he	$+0.134 \pm 0.030$	-0.412 ± 0.029	$+0.546$
`L10.H9.SV0`	1.00	9.15	her, she, She, herself, hers	-0.273 ± 0.041	$+0.652 \pm 0.038$	-0.925
`L11.H7.SV0`	1.3e-5	22.07	herself, Her, her, hers, She	$+0.755 \pm 0.020$	$+0.774 \pm 0.018$	-0.019
`L10.H9.SV1`	0.82	7.87	her, she, herself, She, she	-0.241 ± 0.035	$+0.512 \pm 0.036$	-0.753
`L9.H7.SV0`	1.00	9.23	their, Their, they, their	-0.334 ± 0.028	-0.174 ± 0.031	-0.160
`L6.H3.SV30`	1.00	4.64	they, They, Them, Their	-0.196 ± 0.022	-0.114 ± 0.025	-0.082
`L11.H8.SV23`	0.87	4.32	their, Their, THEY, THEIR, They	-0.278 ± 0.024	-0.143 ± 0.027	-0.135

a compact basis of hard-coded logit receptors that modulate the vocabulary space. These receptors define semantic axes that downstream layers can selectively activate to express context-sensitive meaning. Learning directional masks (via Algorithm 1) further clarifies which of these fixed logit receptors are actually used by the task. Directions with high mask weights may correspond to those carrying discriminative information for the task, while others are effectively pruned. Moreover, it also reflects that some of the inherent structures are not used by a specific dataset, and the discovery of components is highly dependent on the used dataset, which may not contain all the universal inherent directions present in a model.

A notable example is `L11.H7.SV0`, where despite having the largest singular value ($\sigma = 22.07$) and a vocabulary projection strongly associated with feminine tokens, this direction receives an almost-zero mask. Inspection reveals that its activation $\nu_{lh}^\top u_{lhk}$ (a scalar modulating the logit receptor) is nearly identical for male and female examples, essentially encoding variance but no discriminative signal (also see Figure 3). In contrast, directions such as `L10.H9.SV0` show pronounced activation differences across gender contexts and are thus retained. Table 4 summarizes the salient discovered directions, including their mask weights, singular values, top-vocabulary tokens, and conditional activation statistics. Together, these results suggest that the model's internal computation operates through a set of inherently learned logit receptors, fixed basis vectors whose selective activation underlies task behavior and provides a fine-grained, mechanistic bridge between representation and prediction.

This decomposition helps provide an understanding of how the model performs token-level decisions. Each unembedding projection of right-singular direction i.e. $v_{lhk}^\top W_U$ acts as a *fixed* logit receptor controlling the prediction vocabulary space, essentially representing a stable axis associated with a specific token outcome. During inference, the model dynamically steers these receptors through the input-dependent activation coefficients $\nu_{lh}^\top u_{lhk}$. The masks we learn over these singular directions identify which of these receptors the model actually employs, i.e., directions that combine both capacity (large singular value) and discriminative alignment (systematic activation shifts across labels) receive high mask weights, while those that contribute variance but no discriminative signal are suppressed during optimization. As highlighted in Figure 3, the activations $\nu_{lh}^\top u_{lhk}$ form linearly separable clusters for masculine and feminine contexts; directions lacking a decisive separation receive negligible mask weights, indicating that the masking learns to ignore non-discriminative directions. For example, direction `L11.H7.SV0`, despite its large singular value ($\sigma = 22.07$) and a clearly feminine token projection, receives a near-zero mask because its activation distribution is nearly identical across male and female examples.

We further ask what happens if we intervene in these directions. While the above analysis reveals interpretable correlations between singular directions and task behavior, it does not establish whether these directions are *causally responsible* for model predictions. To support our central hypothesis, that transformers distribute computation along distinct, low-rank subfunctions, we require causal evidence that manipulating these subspaces directly alters model output in predictable ways. If a small number of identified singular directions are genuinely responsible for a behavior, then systematically altering them should induce controlled, interpretable changes in the model's output logits, while leaving other computations intact.

Scalar-based Causal Ablation of OV Directions. We therefore perform a series of *Scalar-based counterfactual ablations* designed to test whether the discovered singular directions in the OV projection causally steer gender pronoun prediction. In each ablation, we swap the activations of gender-sensitive singular directions (those with high learned mask values) with the empirically observed mean activations from the opposite gender distribution. For instance, the masculine-sensitive direction `L9.H7.SV1` typically exhibits mean activations of $+0.115$ for "he" prompts and -0.453 for "she" prompts; the intervention exchanges these values, effectively inserting counterfactual gender evidence while keeping all other activations fixed.

Overall, we conduct four controlled experiments: 1) swapping all gender directions for masculine prompts, 2) swapping all gender directions for feminine prompts, 3) swapping only masculine directions for masculine prompts, and 4) swapping only feminine directions for feminine prompts. Each condition measures how average pronoun-logit differences respond to these targeted perturbations, thereby quantifying the causal influence of individual low-rank subspaces.

Formally, for a given singular direction i, the intervention replaces the natural activation $(\nu^\top u_i)$ with the opposite-gender mean activation a'_i:

$$\Delta R = (a'_i - \nu^\top u_i)\, \sigma_i\, v_i^\top .$$

adding this re-scaled vector to the final residual stream vector directly intervenes in the contribution of specific interpretable low-rank features, allowing precise causal editing of gender-specific behavior. Algorithm 2 illustrates this process in more detail. This precisely modifies the residual-stream contribution of selected interpretable features while preserving the model's overall structure, enabling rigorous, scalar-controlled causal testing.

For empirical validation, we consider instances from the GP (Gender Pronoun Resolution) task and segregate prompts into distinct masculine (p_{male}) and feminine (p_{female}) contexts and pass them separately to the model and record the scalar mean corresponding to specific tokens ($\mu_g = a'_{i,g} = \frac{1}{|p_g|}\sum_{x\in p_g} \nu^\top u_i, \quad g \in \{\text{male, female}\}$). This helps provide mean values for each considered context, and for all the pronoun directions $i \in \{$`L9.H7.SV1`, `L11.H8.SV6`, ...$\}$, which are later used in Algorithm 2 to perform intervention and observe the effect of modifying a logit receptor. (also see Figure 2, that illustrates the intervention performed in Algorithm 2)

Findings. The resulting interventions, as highlighted in Table 5, Table 6, and Figure 4, yield a clear causal signal. In Table 5, the Flip→she% is defined as the (# predictions flipped from `'_he'` → `'_she'`) / (# baseline `'_he'` predictions) $\times 100$, and Flip→he% is defined analogously using the baseline `'_she'` predictions, i.e., (# predictions flipped from `'_she'` → `'_he'`) / (# baseline `'_she'` predictions) $\times 100$. Both of these metrics help capture the true flip rate of recoverable predictions (excluding cases where the model initially predicted an "other" token). Across amplification scales shown in both Table 5 and Figure 4, we find that swapping all gender-associated singular directions with their empirically opposite-gender means, and scaling the corresponding singular values by a positive amplification factor, reliably reverses the model's pronoun-logit polarity. For masculine prompts, the mean logit difference shifts from approximately $+2.5$ (favoring `'_he'`) to around -42, and for feminine prompts from $+2.8$ to roughly -41. Moreover, as also reflected in the high flip rates in Table 5, the model almost entirely reverses its prediction given the same baseline context, i.e., prompts containing masculine cues (where the correct baseline prediction is `'_he'`) yield `'_she'` after the mean-swap plus amplification intervention, and symmetrically for feminine-context prompts. The scalar-level ablations in Table 6 further confirm that both full and partial swaps induce substantial, systematic shifts in logit differences, demonstrating that these singular directions contribute additively and consistently to gender-pronoun resolution. All these results provide causal evidence that the identified OV singular directions form *modular, low-rank mechanisms* that directly

Algorithm 2 Intervention for OV Singular Directions

Require: Model $\mathcal{M}$, SVD circuit cache $\mathcal{C}$, data loader $\mathcal{D}$, directions $\mathcal{S}$ (each with $(\ell, h, i, \mu_{\text{he}}, \mu_{\text{she}})$), and target gender $g \in \{\text{he}, \text{she}\}$

Ensure: Summary statistics: mean logit difference, variance, etc.

Initialize results $\leftarrow$ {correct, logit_diff, they_logit}
for each batch (x, y) in $\mathcal{D}$ **do**
  Run model with cache: (logits, cache) $\leftarrow \mathcal{M}$.run_with_cache(x)
  Extract final-token index t^*
  Initialize $\Delta R \leftarrow 0$
  for each $(\ell, h, i, \mu_{\text{he}}, \mu_{\text{she}}) \in \mathcal{S}$ **do**
    Retrieve SVD components $U, S, V \leftarrow \mathcal{C}[\ell, h]$
    Select $(u_i, v_i, \sigma_i) \leftarrow (U[:, i], V[:, i], S[i])$
    Compute attention-weighted context $\nu = [1, \sum \alpha x]$
    Current activation $a_i = (\nu^\top u_i)$
    Target activation $a_i' \leftarrow \begin{cases} \mu_{\text{she}}, & g = \text{he} \\ \mu_{\text{he}}, & g = \text{she} \end{cases}$
    $\Delta a_i \leftarrow (a_i' - a_i)$
    $\sigma_i \leftarrow \sigma_{scale} \times \sigma_i$
    $\Delta R \mathrel{+}= \Delta a_i \, \sigma_i \, v_i^\top$
  end for
  Add ΔR to final residual stream at t^*
  Compute new logits $Z' = \text{LayerNorm}(\text{residual} + \Delta R) W_U + b_U$
end for

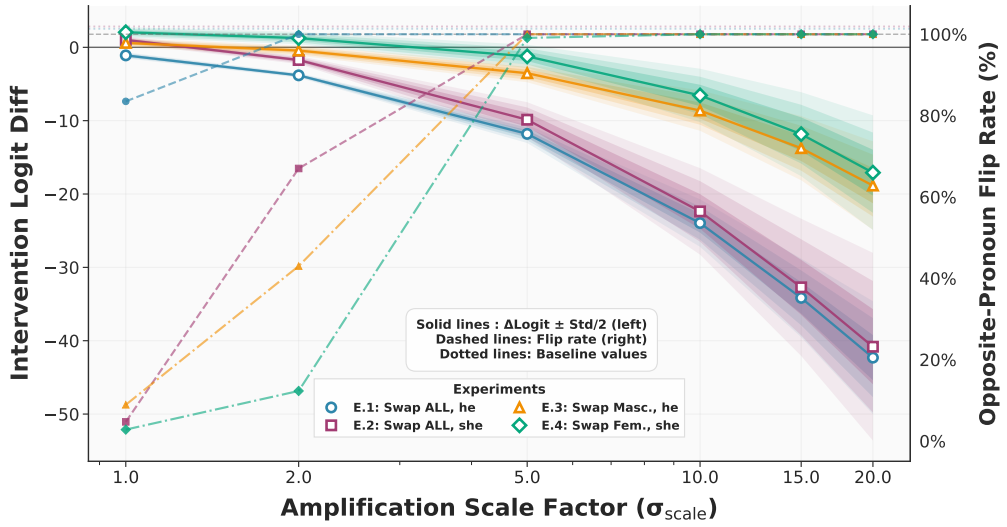

Figure 4: Causal interventions (scaling + swapping) show that singular directions control gender pronoun prediction. The plot displays logit differences (Correct Pronoun − Opposite Pronoun) and flipping rates after intervention. Singular values σ_i are scaled by an integer factor (σ_{scale}) in $\sigma_i(\tilde{a}_i - \nu^\top u_i)v_i^\top$, leading to near-complete prediction reversal at higher scales. This provides causal evidence that these directions are key computational units underlying gender pronoun resolution.

control gender pronoun prediction, demonstrating that interpretable subfunctions are not merely correlational artifacts but causal building blocks of transformer computation, which leads to a new perspective on looking at/understanding predictions made by the model.

We believe that as we move forward, this direction/perspective of decomposing/understanding model decisions will make them more interpretable, and a wider range of transparent conceptualization can be explored/understood in future works.

Table 5: Empirical results of **intervention(scaling + swapping)**. Baseline ΔLogit shows original logit difference between correct and opposite pronoun prediction based on the prompt context. Intervention ΔLogit shows the same after replacing activations of $\nu^\top u$ with respective opposite gender means, and amplifying singular value by multiplying with $\sigma_{\mathbf{scale}}$. Flip rates show the baseline pronoun predictions that were switched to the opposite pronoun by the intervention. All ΔLogit(Logit Difference) values are shown as mean ± std.

Experiment	$\sigma_{\mathbf{scale}}$	**Prompt Type**	**Baseline ΔLogit**	**Interv. ΔLogit**	**Flip→ she%**	**Flip→ he%**
E.1: Swap ALL	1.0	"he"	$+2.53 \pm 1.48$	-1.10 ± 0.83	83.5%	0.0%
E.1: Swap ALL	2.0	"he"	$+2.53 \pm 1.48$	-3.82 ± 0.63	100.0%	0.0%
E.1: Swap ALL	5.0	"he"	$+2.53 \pm 1.48$	-11.80 ± 2.50	100.0%	0.0%
E.1: Swap ALL	10.0	"he"	$+2.53 \pm 1.48$	-23.96 ± 5.52	100.0%	0.0%
E.1: Swap ALL	15.0	"he"	$+2.53 \pm 1.48$	-34.16 ± 7.50	100.0%	0.0%
E.1: Swap ALL	20.0	"he"	$+2.53 \pm 1.48$	-42.31 ± 8.57	100.0%	0.0%
E.2: Swap ALL	1.0	"she"	$+2.84 \pm 2.12$	$+1.01 \pm 0.96$	0.0%	4.7%
E.2: Swap ALL	2.0	"she"	$+2.84 \pm 2.12$	-1.73 ± 0.61	0.0%	67.0%
E.2: Swap ALL	5.0	"she"	$+2.84 \pm 2.12$	-9.87 ± 4.08	33.3%	100.0%
E.2: Swap ALL	10.0	"she"	$+2.84 \pm 2.12$	-22.37 ± 9.31	33.3%	100.0%
E.2: Swap ALL	15.0	"she"	$+2.84 \pm 2.12$	-32.72 ± 13.04	33.3%	100.0%
E.2: Swap ALL	20.0	"she"	$+2.84 \pm 2.12$	-40.82 ± 15.44	33.3%	100.0%
E.3: Swap Masc.	1.0	"he"	$+2.53 \pm 1.48$	$+0.60 \pm 1.22$	8.9%	0.0%
E.3: Swap Masc.	2.0	"he"	$+2.53 \pm 1.48$	-0.44 ± 0.97	43.0%	0.0%
E.3: Swap Masc.	5.0	"he"	$+2.53 \pm 1.48$	-3.53 ± 0.63	100.0%	0.0%
E.3: Swap Masc.	10.0	"he"	$+2.53 \pm 1.48$	-8.60 ± 1.69	100.0%	0.0%
E.3: Swap Masc.	15.0	"he"	$+2.53 \pm 1.48$	-13.50 ± 2.98	100.0%	0.0%
E.3: Swap Masc.	20.0	"he"	$+2.53 \pm 1.48$	-18.15 ± 4.15	100.0%	0.0%
E.4: Swap Fem.	1.0	"she"	$+2.84 \pm 2.12$	$+2.05 \pm 1.39$	0.0%	2.8%
E.4: Swap Fem.	2.0	"she"	$+2.84 \pm 2.12$	$+0.35 \pm 0.73$	0.0%	12.3%
E.4: Swap Fem.	5.0	"she"	$+2.84 \pm 2.12$	-4.75 ± 1.92	16.7%	99.1%
E.4: Swap Fem.	10.0	"she"	$+2.84 \pm 2.12$	-13.04 ± 5.59	50.0%	100.0%
E.4: Swap Fem.	15.0	"she"	$+2.84 \pm 2.12$	-20.73 ± 8.82	50.0%	100.0%
E.4: Swap Fem.	20.0	"she"	$+2.84 \pm 2.12$	-27.59 ± 11.44	50.0%	100.0%

Table 6: Scalar-based ablation of gender-related OV singular directions. Each intervention replaces the scalar activation of each gender direction with its empirical opposite-gender mean and amplifying the singular value($\times 20$ in below table). n is the number of data points

Experiment	**Prompt Context**	n	Baseline ΔLogit	Interv.ΔLogit	Δ(ΔLogit)
E.1: Swap ALL dirs	"he"	150	$+2.53 \pm 1.48$	-42.31 ± 8.57	-44.84
E.2: Swap ALL dirs	"she"	156	$+2.84 \pm 2.12$	-40.82 ± 15.44	-43.66
E.3: Swap Masc. only	"he"	150	$+2.53 \pm 1.48$	-18.15 ± 4.15	-21.38
E.4: Swap Fem. only	"she"	156	$+2.84 \pm 2.12$	-27.59 ± 11.44	-19.93

B.3 Extended Analysis: IOI Task

Additional Mask Visualizations. Figures 5, 6, and 7 provide detailed visualizations of the learned singular value masks for OV, MLP, and QK matrices, respectively. These figures complement the results discussed in the main text by showing head- and layer-level activation patterns across the model. In particular, Figure 5 highlights OV heads with high activation across multiple singular directions, Figure 6 shows layer-wise MLP mask patterns, and Figure 7 quantitatively correlates QK mask values with functional head types identified in prior work [Wang et al., 2022]. These visualizations help provide further evidence that the learned masks capture meaningful functional subcomponents of transformer computation. Figure 8 shows a conceptualization of the SVD-based analysis of Head 9.6, previously identified by Wang et al. [2022] as a "Name Mover" head. The figure

Table 7: The table shows results for the analysis of S_7 attention scores($[1, x_i]\sigma_7 u_7 v_7^\top [1, x_j]^\top$) for head 9.6, highlighting the entity-action distinction role in the IOI task.

Category	Value (Mean $\pm$ SD)	Example tokens	Example values
Entities	$+3.52 \pm 1.42$	"Jerry", "Kevin", "Susan"	$+2.87, +2.29, +2.92$
Actions	-4.44 ± 0.68	"went", "gave", "decided"	$-4.33, -4.17, -5.93$

Table 8: The Table shows results for analysis of S_{28} activations, denoting the role of entity salience in the IOI task.

Token type	Example tokens	SVD_28 values
Named entities (first mention)	"Jerry", "Susan", "Kevin"	5.68, 4.05, 5.22
Named entities (second mention)	"Mary", "Kevin_2", "Marilyn_2"	3.69, 2.07, 2.07
Function words	"the", "to", "of"	0.50, 1.93, 0.15
Discourse connectives	"and"	2.92-3.85

illustrates the combination of multiple distinct functionalities within the QK interaction, including sequence initialization (S1), semantic discrimination (S7), and entity salience (S28), leading to a specific utility in the IOI task.

To complement the in-depth analysis of attention head 9.6 in the main paper, we extend our investigation to two additional heads, 9.9 and 10.0, both previously implicated in the Indirect Object Identification (IOI) task by Wang et al. [2022]. Our aim is to determine whether their internal structure, when decomposed through our singular vector masking framework, reveals similarly modular and interpretable subfunctions. The results provide compelling support for the view that heads encode multiple, independent functional primitives, each realized along distinct low-rank directions.

Head 9.9: Entity Prioritization and Inhibition Dynamics Attention head 9.9 has been previously labeled a "Name Mover" head due to its role in copying names across syntactic spans. However, our singular vector analysis reveals a richer and more nuanced structure. The first and seventh singular directions, S_1 and S_7, consistently exhibit the characteristic pattern of start-of-sequence detectors. These directions assign disproportionately high attention to the first token of each prompt, while suppressing all subsequent tokens. This behavior mirrors similar mechanisms found in other heads and suggests a general strategy for anchoring temporal computations.

More interestingly, the 51^{st} direction, S_{51}, exhibits a selective affinity for named entities, especially those appearing early in a sequence. It systematically favors tokens corresponding to person names while suppressing their repeated mentions. This results in a subtle form of repetition avoidance that biases the model toward novel or contextually salient entities, behavior consistent with entity tracking and attention modulation observed in prior mechanistic studies.

Crucially, S_{16} implements a strong inhibitory signal targeting second mentions of entities. On average, the unnormalized attention score assigned to repeated entities is 2.31 points lower than that assigned to their initial mention, a pattern aligned with the role of S-Inhibition heads described by Wang et al. [2022], which helps suppress the attention. This directional suppression creates a structural preference against re-attending to previously mentioned entities, thereby enhancing the likelihood of selecting an appropriate indirect object. Collectively, these observations demonstrate that head 9.9 multiplexes multiple functions, entity recognition, novelty bias, and inhibition, across orthogonal subspaces.

Head 10.0: Semantic Role Differentiation and IOI Bias Encoding Head 10.0 presents another compelling case of functional modularity. As with head 9.9, the directions S_1 and S_5 operate as strong start-of-sequence indicators, highlighting the recurrence of this structural primitive across layers. However, the sixth singular direction, S_6, shows an especially focused pattern: it consistently assigns high attention scores to named entities and location nouns, while strongly de-emphasizing function words and verbs. For instance, tokens such as "Melissa," "Kelly," and "zoo" receive the highest activations in our sampled prompts, whereas verbs like "gave" and conjunctions like "and" are suppressed. This direction functions as a precise semantic filter, elevating entities and key nouns that anchor the core referential structure of the IOI task.

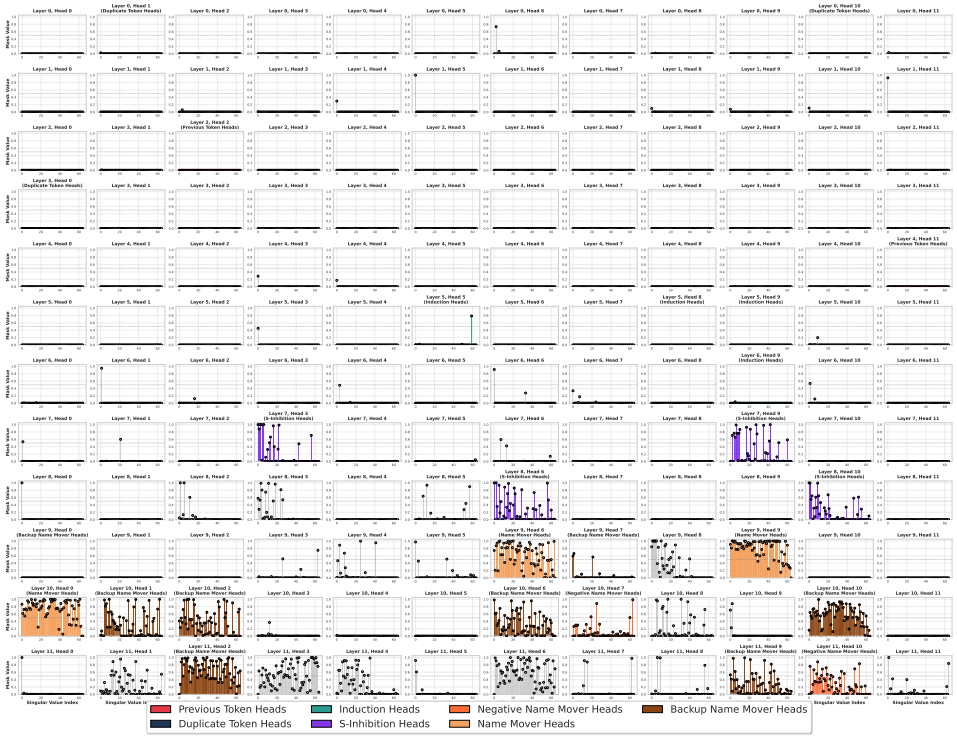

Figure 5: Learned singular value masks for OV ($\mathbf{W}_{\mathbf{aug}}^{\mathbf{OV}}$) matrices across all attention heads in the model. The masks show heads with high activation across multiple singular dimensions correspond to circuit components previously identified by Wang et al. [2022] for the IOI task.

Table 9: Learned mask values and associated functional roles for selected singular directions in `Layer 9, Head 6`. Higher mask values indicate a stronger contribution of that singular direction to the corresponding functional role, illustrating how specific QK directions encode distinct computational subfunctions.

Singular Direction	**Functional Role**	**Learned Mask Value**
S_1	Sequence Initialization Detection	0.53
S_7	Semantic Separation of Entities and Actions	0.64
S_{28}	Entity Salience and Detection	0.97

In contrast, the third direction, S_3, displays a consistent aversion to function words. Words such as "a," "the," and "and" receive markedly negative attention scores, suggesting a broader mechanism of syntactic sparsification. Meanwhile, the seventh direction, S_7, echoes the behavior observed in head 9.6, acting as a semantic separator between entities and verbs. These orthogonal semantic dimensions help disentangle the "who" from the "what," facilitating downstream resolution of coreference.

Finally, the twentieth singular direction, S_{20}, exhibits a statistical preference for indirect objects. In approximately 54–59% of cases, this direction assigns higher attention to the token corresponding to the indirect object, closely mirroring the empirical success rates of the model on the IOI task. This suggests that S_{20} encodes a partial bias that systematically favors the correct grammatical resolution in ambiguous syntactic constructions.

Overall Summary: The analyses of heads 9.9 and 10.0 reinforce our core hypothesis, i.e., transformer components, rather than operating as indivisible functional units, exhibit finely grained internal specialization along interpretable low-rank directions. Each direction performs a targeted role, whether syntactic anchoring, semantic discrimination, or statistical preference modulation, that contributes to the overall behavior of the head. Our results not only validate prior circuit-level findings

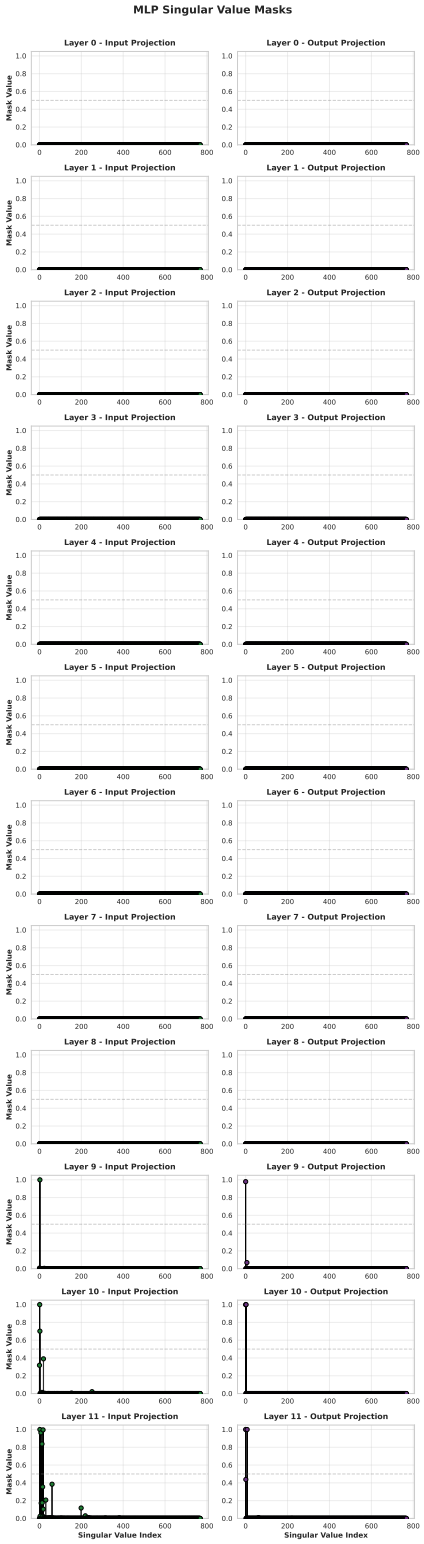

Figure 6: Learned singular value masks for MLP ($\mathbf{W}^{(\mathbf{in})}_{\mathbf{aug}}$(left) and $\mathbf{W}^{(\mathbf{out})}_{\mathbf{aug}}$(right)) matrices across all layers in the model for the IOI task.

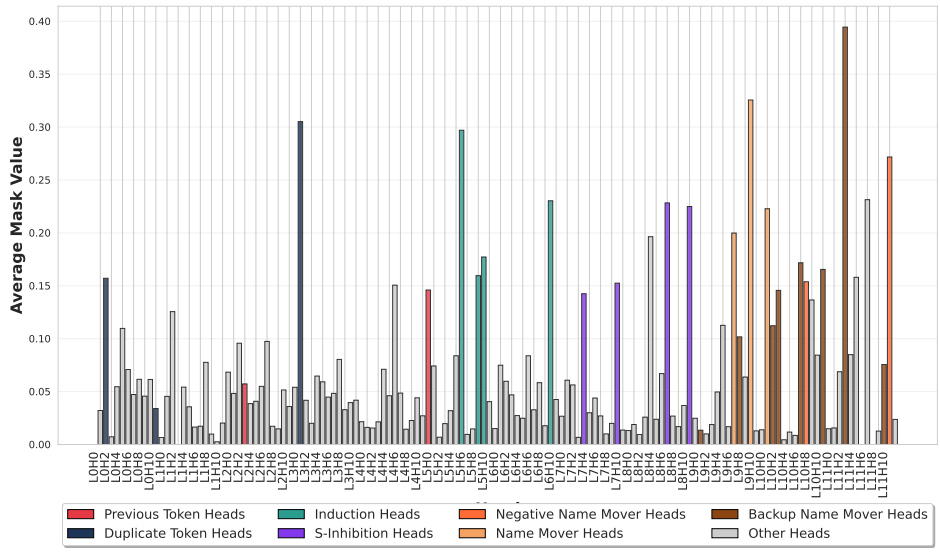

Figure 7: The figure shows the average mask values for W_{aug}^{QK} across attention heads, categorized by functional type. Heads identified in circuits by Wang et al. [2022]; particularly Name Mover, Backup Name Mover, and Negative Name Mover heads, consistently exhibit higher average mask values than other head types. This suggests a correlation between circuit membership and mask activation strength, providing quantitative validation of previously identified functional circuits.

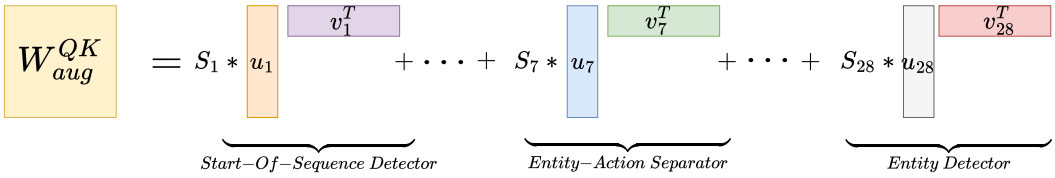

Figure 8: Analysis of head 9.6, previously identified by Wang et al. [2022] as a "Name Mover head" that attends to previous names in a sentence and copies them. Our SVD analysis reveals multiple distinct functionalities within the QK interaction, each serving specific roles consistently across the dataset. This decomposition provides a more nuanced understanding of the head's behavior.

but also refine them by isolating the precise mechanisms responsible for observed model behavior. This decomposition presents a powerful perspective for interpretability, enabling a move from coarse component-level attributions to direction-level mechanistic understanding.

B.4 Analysis for *Greater Than* task

To evaluate the generality of our method beyond the IOI task, we extend our analysis to the *Greater Than* benchmark introduced by Hanna et al. [2023], which investigates a model's capacity for numerical comparison. Each input prompt presents two years within a templated sentence, e.g., *"The treaty lasted from the year 1314 to the year 13"*, and the model must complete the final token(s) such that the resulting year is strictly greater than the first. Crucially, the completion must yield a valid multi-token year (e.g., **28** completing "1328"), with careful curation to avoid boundary conditions and single-token years that could confound interpretability.

We analyze this task by focusing on attention heads previously identified as critical to the model's numerical comparison behavior, specifically heads 6.9, 9.1, and 5.5 in GPT-2. Using our method, we dissect the W_{aug}^{QK} matrices of these heads into their dominant singular directions and interpret their respective contributions.

Table 10: SVD Component Analysis of $\mathbf{W}_{\text{aug}}^{\mathbf{QK}}$ Attention for the *Greater Than* Task , Focus on Head 9.1. Multiple low-rank components exhibit high attention to the target year token (YY), which is crucial for accurate prediction. The "Highest Attention %" indicates how often YY received the highest attention score.

SVD Component	Avg. Attention ($\pm$ Std)	Highest Attention (%)	Mask Value
S_{31}	7.09 ± 1.68	100	1.00
S_3	4.79 ± 1.08	99.8	1.00
S_{26}	1.86 ± 0.87	90.9	3.29×10^{-5}
S_{59}	2.94 ± 1.86	81.8	1.00
S_{37}	4.71 ± 2.18	81.5	1.00
S_{32}	3.90 ± 1.65	72.7	1.00

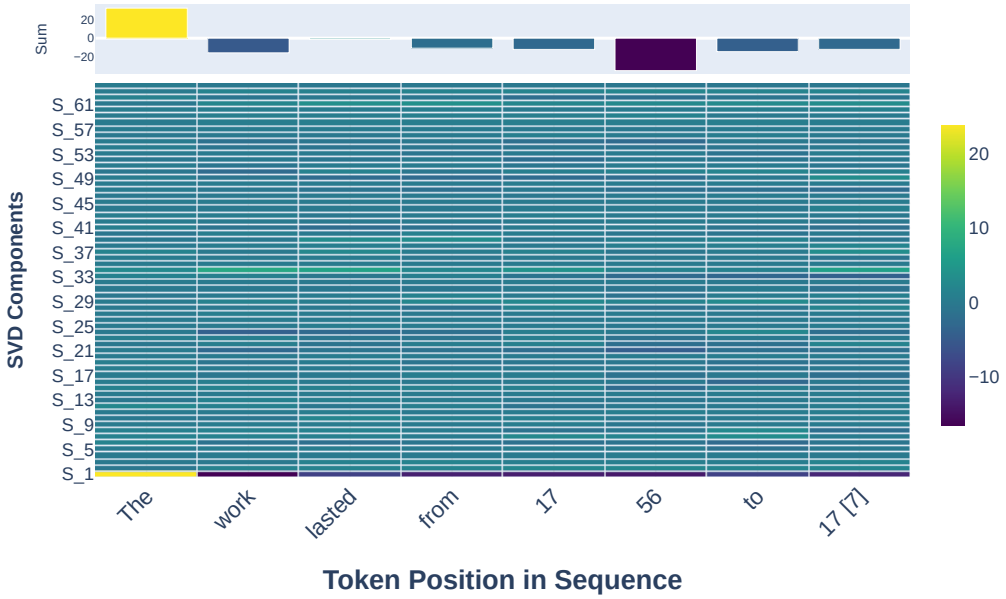

Figure 9: The above figure shows the attention score of head 9.1 for the *Greater Than* task. S_1 attends highly to the first token, and others, such as S_3 and S_{31}, predominantly focus on the end-of-year token.

Head 9.1: Distributed Temporal Anchoring Attention head 9.1 (Figure 9) reveals a distributed mechanism for endpoint detection. The first singular direction (S_1) consistently assigns high positive attention to the start-of-sequence token, while giving negative scores to subsequent ones. This behavior functions as a positional prior, commonly reused across tasks.

More importantly, multiple directions contribute to highlighting the terminal year token (YY). For instance, S_{31} and S_3 achieve near-perfect precision in identifying YY as the attention focus, doing so in 100% and 99.8% of prompts, respectively (see Table 10). Other components such as S_{32} and S_{59} also consistently amplify YY, although with lower magnitude or frequency. This suggests that head 9.1 does not rely on a single dominant axis, but instead employs a compositional mechanism where multiple orthogonal components redundantly encode attention to temporal anchors.

Table 11: SVD Component Analysis of $\mathbf{W}_{\text{aug}}^{\mathbf{QK}}$ Attention for the *Greater Than* Task , Focus on Head 6.9. Multiple low-rank components exhibit high attention to the target year token (YY), which is crucial for accurate prediction. The "Highest Attention %" indicates how often YY received the highest attention score.

SVD Component	Avg. Attention $\pm$ Std	Highest Attention (%)	Mask Value
S_2	10.747 ± 2.858	100.0	1.00
S_7	4.324 ± 2.482	83.3	6.81×10^{-6}
S_{18}	4.680 ± 2.327	83.3	0.15
S_{20}	4.897 ± 4.967	66.7	0.99
S_{29}	3.249 ± 3.587	50.0	1.14×10^{-5}

Head 6.9: Sharply Localized Numerical Discrimination In contrast, head 6.9 demonstrates highly concentrated behavior. The second singular direction, S_2, exhibits extremely sharp selectivity, consistently assigning the highest attention to the final year token with 100% accuracy ($\mu = 10.747 \pm 2.858$) (see Table 11 for reference). This direction appears to isolate the second numeric entity in the input, crucial for the Greater Than judgment. Supporting directions such as S_7, S_{18}, and S_{20} reinforce this emphasis with selection rates above 66%, albeit at lower attention magnitudes. This pattern reflects a localized, narrowly targeted strategy for operand comparison.

Head 5.5: Redundant High-Magnitude in Reinforcing the Endpoint Head 5.5 features a richer distribution of strong attention-inducing components. The leading singular direction (S_1) again exhibits the start-of-sequence pattern, while several others like S_{47}, S_{28}, S_{52}, robustly identify the YY token. Notably, S_{15} stands out with the highest attention magnitude across all heads (19.14 ± 7.15), suggesting an exceptionally focused mechanism for endpoint amplification.

While some of these components exhibit redundancy in their selection behavior (e.g., S_{28} and S_{52} both achieving 87.5% attention on YY), they differ significantly in their mean scores and variance. This dispersion implies an ensemble encoding, where multiple vectors provide convergent evidence toward the correct numeric endpoint, increasing the model's reliability across input variations.

Cross-Head Trends and Shared Functionalities Across all heads, we observe the emergence of two recurring functional patterns: (1) start-of-sequence structure detection via S_1, and (2) endpoint amplification via a sparse set of highly specialized directions. These shared functionalities suggest that certain subspaces, such as those capturing positional priors or token-type distinctions (e.g., numbers vs. nouns), may be reused across attention contexts. Moreover, the existence of multiple low-rank directions targeting the same token suggests that GPT-2 distributes the implementation of numerical reasoning across a sparse ensemble of interpretable basis vectors.

Overall Summary: This decomposition of the Greater Than task reveals a consistent, structured strategy whereby numerical comparison is encoded not in monolithic head-level behavior, but in sparse, orthogonal subspaces within each attention matrix. Each direction contributes a distinct yet complementary role, which includes establishing context, identifying operand positions, and assigning salience to temporally relevant entities. These findings provide compelling evidence that transformers perform symbolic reasoning via emergent low-dimensional structures, and that these structures are modular, reusable, and interpretable via singular decomposition.

B.5 Universal Composite Functionalities Discovered

Our analysis reveals that certain singular value components exhibit consistent, reusable functionalities across different attention heads and tasks, pointing to the presence of universal composite functionalities within GPT-2's internal representations. For example, component S_1 in attention head 9.6 consistently acts as a *start-of-sequence detector*, robustly assigning high attention scores to the initial token regardless of the task context, as illustrated in Figure 10. This persistent role suggests that some attention subspaces are dedicated to foundational structural signals critical for sequence processing. Similarly, component S_7 of the same head functions reliably as an *entity-action separator*, selectively attending to name and object entities while suppressing attention to action or functional tokens, as shown in Figure 11. The consistency of these components across diverse

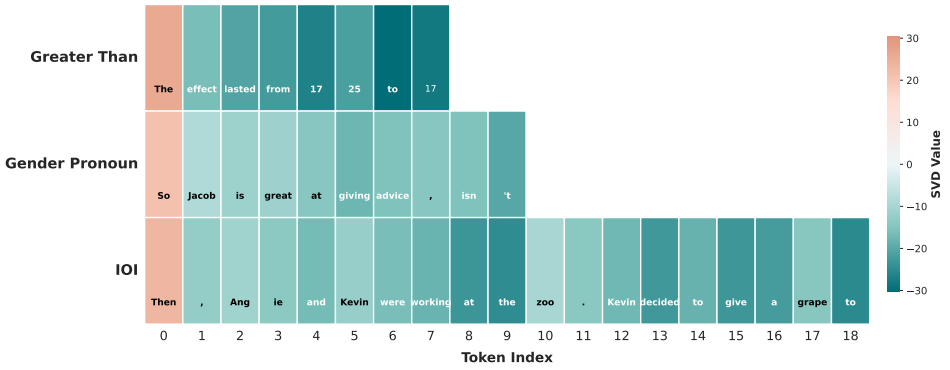

Figure 10: The figure above shows an instance of attention scores of the final token for component S_1 of head 9.6. It demonstrates that S_1 consistently functions as a start-of-sequence detector across tasks, independent of context.

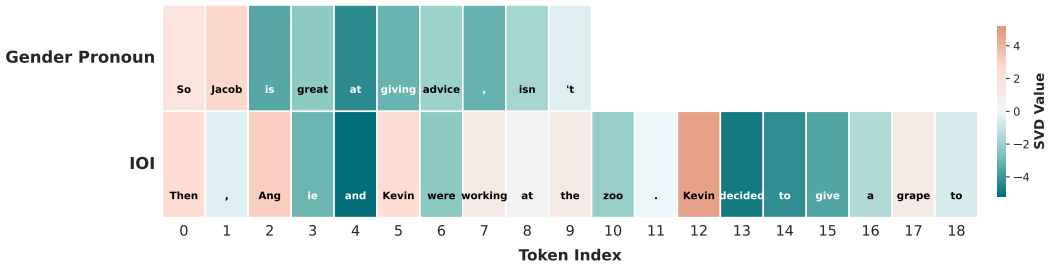

Figure 11: The figure above shows an instance of attention scores of the final token for component S_7 of head 9.6. It demonstrates that S_7 consistently act as an Entity action separator by giving the highest attention score to Name and object entities, and the least attention score to actions.

contexts implies that transformer models implement a set of primitive, composable mechanisms that are repurposed modularly to support a variety of reasoning and linguistic tasks. Uncovering and characterizing such universal functionalities not only enriches our understanding of transformer interpretability but also paves the way for targeted interventions, modular editing, and transfer of learned behaviors across models and domains. These findings motivate further investigation into other universal components and their interplay, which may reveal a hierarchical structure of model reasoning primitives embedded in singular value decompositions of attention matrices.

B.6 Sparsity Computation

We define sparsity in terms of the number of singular directions that are effectively suppressed during the mask learning process. Prior to optimization, all directions with negligible singular values are discarded, leading to a negligible penalty of approximately 10^{-6} in KL divergence. The remaining directions constitute the learnable subspace over which mask optimization is performed.

Since the model undergoes two kinds of compression, (i) Zeroing out directions with near-zero singular values, and (ii) Pruning directions with high singular values through learned masks, we report two complementary sparsity measures:

Relative Sparsity. Let n_{active} denote the number of singular directions having greater mask value then the $threshold(1 \times 10^{-2})$ after mask optimization, and $N_{\text{learnable}}$ the total number of directions subject to training. Then the relative sparsity is

$$S_{\text{rel}} = 1 - \frac{n_{\text{active}}}{N_{\text{learnable}}}.$$

This measures sparsity within the learnable subset only.

Full Sparsity. Considering the complete model, let N_{total} be the total number of singular directions available across all OV projections. The full sparsity is defined as

$$S_{\text{full}} = 1 - \frac{n_{\text{active}}}{N_{\text{total}}}.$$

This reflects overall model compression after both truncation and pruning.

B.7 Discussion and Future Directions

Our proposed decomposition-based framework presents a scalable, model-agnostic approach for discovering interpretable subspaces within pretrained transformers. By applying singular value decomposition to attention weight matrices, we identify low-rank directions that robustly encode functional roles, such as temporal endpoint detection or entity repetition suppression. This enables fine-grained dissection of emergent behaviors, such as the "Name Mover" or "Greater Than" circuits, and reveals that these are implemented through sparse combinations of reusable subspaces rather than monolithic structures. Notably, we observe recurring components across tasks, such as start-of-sequence detectors and temporal amplifiers, suggesting that transformer models like GPT-2 leverage a compact set of primitive functions in compositionally rich ways. Our results align with and extend prior mechanistic interpretability research, complementing methods such as contextual decomposition [Hsu et al., 2025], attribution patching [Nanda et al., 2023], and circuit overlap metrics [Hanna et al., 2024], while providing a new axis of interpretability rooted in linear decomposability. The approach is computationally efficient, all analyses were performed on a single NVIDIA A40 GPU, and it reveals that many key behaviors are localized to a small number of interpretable singular directions. This raises intriguing possibilities for circuit editing, model compression, or steering via low-rank intervention. However, our results are not exhaustive; rather, they provide a first step toward uncovering modular internal mechanisms through decomposition. We emphasize that this work identifies functionalities that are interpretable under this perspective, but there is much more to uncover. Future directions include conducting targeted intervention experiments to validate causal contributions of discovered components, examining whether these mechanisms generalize to larger or more capable models (e.g., Phi, LLaMA), and exploring their activation in real-world settings. Additionally, the search for composite patterns, i.e., higher-order patterns composed of recurring singular directions, may yield deeper insight into how transformers orchestrate symbolic reasoning.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We provide evidence for claims in Section 2, Section 3, Section 4, and Section 5.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Limitations sub-section is included in Section 7

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: We provide the derivation in Section 2.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide details in Section 4, Section 5, and Appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We will release model and experiment code upon acceptance.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Details are provided in Section 4, Section 5, and Appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Details are provided in Section 4 and Appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Details are provided in Appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have reviewed code of ethics and have abided by them.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: In this work, we do not develop any new model or technology but perform analysis of existing pre-trained models. To the best of our knowledge there is no negative societal impact.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: We are not creating any new datasets or models.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We attribute the creators of various models in various sections of the paper and in the appendix.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: We are not creating any new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: We do not perform any human experiments.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: We do not perform any human experiments.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: This paper is about analysis of inner-workings of LLMs and we have described their usage in detail in several sections of the paper.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.