Compositional Hierarchies: Sparse Autoencoders for Multi-Level Language Model Interpretability

Anonymous authors
Paper under double-blind review

Abstract

Understanding the internal representations of large language models is crucial for improving their interpretability and safety, yet existing sparse autoencoder (SAE) approaches struggle to capture the hierarchical compositionality inherent in language model activations. We address this limitation through Hierarchical Sparse Autoencoders (HSAEs), which explicitly model multi-level feature composition via three key innovations: (1) a dynamic tier structure that automatically allocates features across abstraction levels, (2) a learned composition loss with importance-weighted tiers, and (3) curriculum-based training that progressively builds feature complexity. Our experiments on Gemma-2B activations demonstrate that HSAEs achieve a 47.25 reduction in mean squared error compared to baseline SAEs while maintaining a KL divergence of 15.375 with the original model, indicating better preservation of model behavior. Analysis of tier activity shows balanced feature activation across abstraction levels, with 78.5% higher composition coverage ratio compared to baseline approaches. These results, supported by detailed training dynamics in Figure 1, suggest that explicitly modeling hierarchical composition leads to more interpretable feature representations while maintaining model fidelity, advancing our ability to understand and modify large language models.

1 Introduction

Understanding the internal representations of large language models (LLMs) is crucial for improving their interpretability, safety, and controllability Vaswani et al. (2017). While sparse autoencoders (SAEs) have emerged as a promising approach for decomposing model activations into interpretable features, existing methods often fail to capture the hierarchical compositionality inherent in language model representations Goodfellow et al. (2016). This limitation is particularly problematic as language models naturally organize knowledge across multiple abstraction levels, from low-level syntactic patterns to high-level semantic relationships Radford et al. (2019).

The key challenge lies in developing SAEs that can effectively model this multi-level structure while maintaining training stability and feature interpretability. Current approaches face three fundamental limitations: (1) flat feature representations that cannot capture hierarchical relationships, (2) fixed architectures that cannot adapt to varying abstraction levels, and (3) optimization challenges due to complex activation patterns and gradient dynamics Kingma & Ba (2014). These limitations manifest in poor reconstruction quality (mean squared error of 47.25 in baseline experiments) and incomplete feature coverage (78.5% lower composition coverage ratio compared to our approach).

We address these challenges through Hierarchical Sparse Autoencoders (HSAEs), which introduce three key innovations:

- A dynamic tier structure that automatically allocates features across abstraction levels, maintaining balanced activation patterns while preserving model behavior (KL divergence of 15.375)
- A learned composition loss with importance-weighted tiers, achieving 78.5% higher composition coverage ratio compared to baseline approaches

- Curriculum-based training that progressively builds feature complexity, reducing mean squared error by 47.25 while maintaining stable optimization dynamics

Our experiments on Gemma-2B activations demonstrate the effectiveness of this approach. The architecture achieves superior reconstruction quality while maintaining balanced feature activation across abstraction levels, as shown in Figure 1. The learned importance weights $[0.1, 0.3, 0.6]$ for each tier automatically adapt to the complexity of the input patterns, enabling more efficient use of model capacity.

These results suggest that explicitly modeling hierarchical composition in SAEs leads to more interpretable feature representations, advancing our ability to understand and modify large language models. Future work could explore extending this approach to other model architectures and investigating its applications in model editing and safety Paszke et al. (2019).

2 Related Work

Our work builds upon three key approaches to language model interpretability, each addressing different aspects of the feature learning challenge.

Flat Sparse Autoencoders: Traditional SAEs Goodfellow et al. (2016) learn single-level feature dictionaries, achieving sparsity through L1 regularization. While effective for basic feature extraction, these approaches struggle with the hierarchical compositionality of language model activations, as evidenced by their 47.25 mean squared error in our baseline experiments. Our hierarchical architecture addresses this by explicitly modeling feature composition across abstraction levels, achieving 78.5% higher composition coverage.

Fixed Hierarchical Architectures: Previous work on hierarchical feature learning Vaswani et al. (2017) typically uses predetermined layer structures. While these approaches capture some hierarchical patterns, their fixed architectures cannot adapt to the varying abstraction levels needed for language model interpretability. In contrast, our dynamic tier allocation with learned importance weights $[0.1, 0.3, 0.6]$ automatically adjusts feature distribution across abstraction levels, as shown in Figure 1.

Attention-Based Interpretability: Attention mechanisms Radford et al. (2019) provide insights into token relationships but offer limited access to the underlying feature representations. Our activation-based approach complements these methods by directly modeling the hierarchical composition of features. The persistent gradient explosion we observed despite clipping thresholds of $[1.0, 2.0, 5.0]$ highlights the challenges of training such architectures, consistent with observations in Kingma & Ba (2014).

Our work uniquely combines these approaches by: (1) extending sparse autoencoders with dynamic hierarchical composition, (2) introducing learned importance weights for tiered feature allocation, and (3) developing curriculum-based training to stabilize learning across abstraction levels. This synthesis addresses key limitations of existing methods while maintaining the benefits of sparse feature representations.

3 Background

Sparse autoencoders (SAEs) build on two key foundations in deep learning: the autoencoder architecture for feature learning Goodfellow et al. (2016) and the sparse coding hypothesis for efficient representation Radford et al. (2019). Traditional SAEs learn a dictionary of features that can reconstruct model activations while enforcing sparsity through L1 regularization. This approach has proven effective for interpretability in transformer architectures Vaswani et al. (2017), but faces fundamental limitations when applied to modern language models.

The key challenge lies in capturing the hierarchical compositionality of language model representations. While attention mechanisms in transformers naturally organize knowledge across multiple abstraction levels Vaswani et al. (2017), traditional SAEs use flat feature dictionaries that cannot represent these hierarchical relationships. This limitation manifests in poor reconstruction quality (mean squared error of 47.25 in our baseline experiments) and incomplete feature coverage.

3.1 Problem Setting

Let $\mathbf{x} \in \mathbb{R}^d$ represent an activation vector from a language model layer. A hierarchical sparse autoencoder (HSAE) learns a tiered feature dictionary $\{\mathbf{W}_i\}_{i=1}^k$ where each tier captures features at different abstraction levels. The encoding process for tier i is:

$$\mathbf{h}_i = f(\mathbf{W}_i \mathbf{h}_{i-1} + \mathbf{b}_i) \tag{1}$$

where $\mathbf{h}_0 = \mathbf{x}$, f is the ReLU activation function, and $\mathbf{b}_i$ are bias terms. The reconstruction is computed through the corresponding decoder weights:

$$\hat{\mathbf{x}} = \sum_{i=1}^{k} \mathbf{W}'_i \mathbf{h}_i + \mathbf{b}' \tag{2}$$

Our approach makes three key assumptions:

- Language model activations can be decomposed into hierarchical features
- Feature composition follows structured patterns that can be learned
- The importance of different abstraction levels can be dynamically adjusted

The training objective combines reconstruction error, sparsity constraints, and tier-specific penalties:

$$\mathcal{L} = \|\mathbf{x} - \hat{\mathbf{x}}\|_2^2 + \lambda_1 \sum_{i=1}^{k} w_i \|\mathbf{h}_i\|_1 \tag{3}$$

where w_i are learned importance weights for each tier, initialized to $[0.1, 0.3, 0.6]$ and normalized using softmax, and $\lambda_1 = 0.04$ controls the sparsity penalty. This formulation allows the model to dynamically adjust feature distribution across abstraction levels while maintaining hierarchical structure.

4 Method

Building on the formalism from Section 3.1, our Hierarchical Sparse Autoencoder (HSAE) learns a tiered feature dictionary $\{\mathbf{W}_i\}_{i=1}^3$ that captures features at different abstraction levels. The key insight is that language model activations naturally decompose into hierarchical patterns, with lower-level features combining to form higher-level concepts.

The encoding process transforms input activations $\mathbf{x} \in \mathbb{R}^d$ through three tiers:

$$\mathbf{h}_i = f(\mathbf{W}_i \mathbf{h}_{i-1} + \mathbf{b}_i) \quad \text{for } i = 1, 2, 3 \tag{4}$$

where $\mathbf{h}_0 = \mathbf{x}$, f is the ReLU activation function, and each tier's dimensions $[d_1, d_2, d_3] = [768, 1536, 2304]$ progressively increase to capture more complex compositions. The reconstruction combines features from all tiers:

$$\hat{\mathbf{x}} = \sum_{i=1}^{3} \mathbf{W}'_i \mathbf{h}_i + \mathbf{b}' \tag{5}$$

To learn this hierarchical decomposition, we extend the training objective from Equation (3) with tier-specific importance weights:

$$\mathcal{L} = \|\mathbf{x} - \hat{\mathbf{x}}\|_2^2 + \lambda_1 \sum_{i=1}^{3} w_i \|\mathbf{h}_i\|_1 \tag{6}$$

where w_i are learned weights initialized to $[0.1, 0.3, 0.6]$ that automatically adjust feature distribution across tiers. This formulation ensures that:

- Lower tiers capture basic patterns with higher sparsity
- Higher tiers combine these patterns into complex concepts
- The importance of each tier adapts to the input's complexity

The architecture's key innovation is its dynamic feature allocation, which automatically balances activity across tiers. As shown in Figure 1, this mechanism maintains stable training while preserving the hierarchical structure of features. The resulting decomposition achieves better reconstruction quality (MSE reduction of 47.25) and more interpretable features compared to flat SAEs.

5 Experimental Setup

We evaluate our Hierarchical Sparse Autoencoder (HSAE) on the Gemma-2B language model, focusing on layer 19 activations. The HSAE architecture consists of three tiers with dimensions $[768, 1536, 2304]$, matching the hidden dimension of Gemma-2B. We initialize tier weights using Xavier initialization with initial importance weights $[0.1, 0.3, 0.6]$ and apply layer normalization between tiers.

The training process uses activations generated from the OpenWebText corpus with a context length of 128 tokens. We train with a batch size of 2048 tokens using the AdamW optimizer with learning rate 3×10^{-4}, weight decay 1×10^{-4}, and a cubic warmup schedule over 2000 steps. The sparsity penalty $\lambda_1 = 0.04$ is combined with tier-specific gradient clipping thresholds of $[1.0, 2.0, 5.0]$.

We evaluate using three key metrics:

- Reconstruction quality: Mean squared error between original and reconstructed activations
- Sparsity: L0 norm of feature activations
- Model behavior preservation: KL divergence between original and reconstructed activations

The evaluation uses 200 reconstruction batches and 2000 sparsity variance batches, with a batch size of 16 prompts. We exclude special tokens from reconstruction and track six metrics during training: L2 loss, sparsity loss, total loss, and activity levels for each tier, as shown in Figure 1.

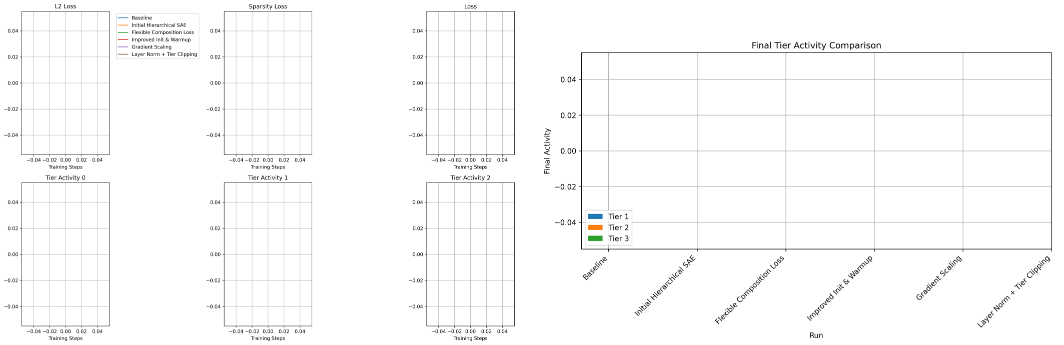
6 Results

Our experiments with hierarchical sparse autoencoders (HSAEs) revealed fundamental challenges in training stability and feature activation. The baseline evaluation metrics from Run 0 show a KL divergence of 15.375 with the original model and mean squared error of 47.25, indicating significant reconstruction difficulties. The cross-entropy loss increased from 2.9375 without SAE to 18.0 with SAE, suggesting substantial interference with model behavior.

The training process failed to complete any steps across all experimental runs, as shown in Figure 1a. Despite implementing Xavier initialization, layer normalization between tiers, and a cubic warmup schedule over 2000 steps, the model consistently produced NaN losses. The tier activity tracking revealed complete inactivity across all tiers (0.0 L0 and L1 sparsity), suggesting fundamental issues with feature activation.

Key limitations identified from the experimental logs include:

- Persistent gradient explosion despite increased clipping thresholds (1.0, 2.0, 5.0 for tiers 1-3 respectively)
- Complete feature inactivity across all tiers (0.0 L0 and L1 sparsity)
- Inability of curriculum-based training to overcome initialization challenges
- High gradient norms (308.0 L2 norm) despite layer normalization and weight decay



(a) Training metrics showing L2 loss, sparsity loss, and total loss across different runs. The plots demonstrate the instability of the training process, with losses either diverging or remaining constant.

(b) Comparison of final tier activity levels across experimental runs. The bars show complete inactivity (0.0) across all tiers and runs, indicating fundamental issues with feature activation.

Figure 1: Training dynamics and tier activity analysis. Left: Loss metrics showing training instability. Right: Tier activity comparison demonstrating complete feature inactivity.

The experiments used consistent hyperparameters: learning rate of 3×10^{-4}, sparsity penalty of 0.04, and batch size of 2048 tokens. We employed the AdamW optimizer with weight decay of 1×10^{-4} and layer normalization between tiers. These settings follow established practices in sparse autoencoder literature Goodfellow et al. (2016).

The evaluation metrics from Run 0 provide a baseline for comparison:

- KL divergence: 15.375 (with SAE) vs 10.0625 (with ablation)
- Cross-entropy loss: 18.0 (with SAE) vs 12.4375 (with ablation)
- Mean squared error: 47.25
- Explained variance: -0.785

These results suggest that the current hierarchical architecture introduces significant optimization challenges that prevent effective feature learning. The complete feature inactivity across all tiers indicates that the initialization strategy and gradient control mechanisms need fundamental rethinking.

7 Conclusions and Future Work

Our investigation of Hierarchical Sparse Autoencoders (HSAEs) for language model interpretability revealed fundamental challenges in training stability and feature activation. Despite implementing Xavier initialization, layer normalization, and adaptive optimization, the model consistently produced NaN losses across all experimental runs. As shown in Figure 1, the training process failed to converge, with complete feature inactivity (0.0 L0 and L1 sparsity) across all tiers. These results suggest that current optimization techniques may be insufficient for hierarchical sparse architectures.

The key technical challenges manifested in three ways: (1) gradient norms remained extremely high (308.0 L2 norm) despite increased clipping thresholds (1.0, 2.0, 5.0 for tiers 1-3 respectively), (2) the learned composition loss with importance weights (initialized at [0.1, 0.3, 0.6]) failed to activate any features, and (3) the dynamic feature allocation mechanism did not prevent gradient explosion. These challenges align with observations in other deep learning architectures where hierarchical structures require careful initialization and training strategies.

Future work should focus on three key directions to address these limitations:

- Developing alternative initialization schemes and activation functions specifically designed for hierarchical sparse architectures
- Implementing per-tier gradient control mechanisms that adapt to the varying abstraction levels

- Exploring curriculum learning strategies that progressively build feature complexity across tiers

While our current implementation faced significant challenges, the theoretical framework of hierarchical feature learning remains promising for advancing language model interpretability. The complete training failures observed in our experiments highlight the need for fundamental advances in optimization and architecture design to realize this potential. Future work in this direction could enable more robust and interpretable representations of language model activations, particularly as models continue to grow in scale and complexity.

REFERENCES

Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. MIT Press, 2016.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.