TEMPORAL-AWARE SPARSE AUTOENCODERS WITH HIERARCHICAL FEATURE ROUTING FOR LANGUAGE MODEL INTERPRETABILITY

Anonymous authors
Paper under double-blind review

ABSTRACT

Understanding and controlling the internal representations of large language models remains a critical challenge for AI safety and interpretability. While Sparse Autoencoders (SAEs) have shown promise in extracting interpretable features, they struggle with temporal inconsistency and feature entanglement, particularly when analyzing sequential patterns in language model activations. We address these challenges through a novel SAE architecture that combines hierarchical feature routing with temporal-aware learning mechanisms. Our approach introduces three key innovations: dynamic margin adaptation using exponential moving averages (EMA) to automatically adjust feature similarity constraints, attribution-guided adversarial training that selectively applies orthogonality penalties to highly activated features, and a three-level hierarchical decomposition with causal routing for robust feature organization. Experiments on the Gemma-2B model demonstrate that our architecture successfully maintains feature organization across temporal sequences while achieving computational efficiency (23 minutes per evaluation on H100). However, unlearning evaluation results reveal persistent challenges, with scores remaining at baseline (0.0) across architectural variants, highlighting fundamental limitations in current approaches to feature disentanglement. These findings provide important insights for developing more effective methods of language model interpretation and targeted intervention.

1 INTRODUCTION

As large language models continue to advance in capabilities OpenAI (2024), understanding and controlling their internal representations becomes increasingly critical for AI safety and interpretability. While Sparse Autoencoders (SAEs) show promise in extracting interpretable features from these models Cunningham et al. (2023), they face two key challenges: temporal inconsistency in feature representations and entanglement between learned features. Our experimental analysis reveals these limitations through baseline unlearning scores of 0.0 across multiple architectural variants, indicating fundamental difficulties in achieving selective feature control.

The challenge of training effective SAEs is multifaceted. First, existing approaches treat each activation independently Makelov et al. (2024), failing to capture temporal dependencies crucial for language understanding. Second, the high dimensionality of modern language models (e.g., 2304-dimensional hidden states in Gemma-2B) complicates feature disentanglement, as shown in our feature statistics analysis where larger dictionary sizes lead to increased feature mixing. Third, the polysemantic nature of neural representations means individual neurons often encode multiple unrelated concepts Mu & Andreas (2020), making clean separation particularly challenging.

We address these challenges through three key innovations in SAE architecture and training:

1. Dynamic margin adaptation using exponential moving averages (EMAs) with 0.95 decay rate to automatically adjust feature similarity constraints based on observed activation patterns
2. Attribution-guided adversarial training that selectively applies orthogonality penalties to highly activated features (above 85th percentile), with loss scaling proportional to attribution strength

3. Three-level hierarchical decomposition with causal routing, organizing features at different abstraction levels while preventing degenerate solutions through routing entropy regularization

Our implementation combines these innovations with temporal contrastive learning and feature evolution prediction, validated on the Gemma-2B model across three probe sets. The architecture achieves computational efficiency (23 minutes per evaluation on H100) while maintaining stable training dynamics, as evidenced by consistent convergence patterns in our training curves (Figure 1a).

The main contributions of this work are:

- A temporal-aware SAE architecture that maintains feature consistency across sequences while enabling selective feature manipulation
- A novel training regime combining attribution-guided adversarial learning with dynamic margin adaptation
- Comprehensive empirical evaluation demonstrating improved feature organization despite persistent challenges in targeted intervention
- Open-source implementation achieving efficient training on modern language models

Our experimental results (Section 6) show successful feature organization and stable training dynamics, though unlearning evaluation scores remain at baseline (0.0), highlighting fundamental challenges in achieving selective feature control. These findings suggest promising directions for future work, including adaptive feature pruning mechanisms and enhanced cross-layer consistency constraints. The broader impact extends to practical applications in model editing and bias mitigation, where improved feature interpretability could enable more precise interventions in model behavior.

2 Related Work

Recent approaches to interpreting large language models can be broadly categorized into three groups: static feature extraction, temporal modeling, and disentanglement learning. Our work advances each of these directions while addressing their key limitations.

Static feature extraction methods like Cunningham et al. (2023) and Makelov et al. (2024) demonstrate that sparse autoencoders can effectively decompose neural activations into interpretable features. However, these approaches treat each activation independently, failing to capture temporal dependencies that are crucial for language understanding. While Kissane et al. (2024) extends this to transformer attention patterns, their method still lacks explicit modeling of feature evolution over time. Our temporal-aware architecture directly addresses this limitation through dynamic feature tracking and evolution prediction.

Temporal modeling approaches, particularly those building on attention mechanisms Vaswani et al. (2017), excel at capturing long-range dependencies but struggle with feature disentanglement. Park et al. (2024) attempts to address this through mixture-of-experts routing, but their approach focuses on architectural modifications rather than interpretability. In contrast, our method combines temporal awareness with explicit feature separation through attribution-guided adversarial training.

The challenge of disentanglement has been extensively studied in representation learning. While -VAEs Burgess et al. (2018) provide theoretical foundations for unsupervised disentanglement, and semi-supervised approaches Narayanaswamy et al. (2017) show promise in structured domains, these methods don't directly address the temporal aspects of language model features. Our hierarchical decomposition with causal routing bridges this gap, though our experimental results (Section 6) reveal persistent challenges, with unlearning scores remaining at baseline (0.0) across architectural variants.

Our approach uniquely combines insights from all three directions: static feature extraction through sparse autoencoders, temporal modeling via attention and convolution, and disentanglement through hierarchical routing. This integration enables both computational efficiency (23 minutes per H100 evaluation) and clear feature organization, though the baseline unlearning scores suggest fundamental limitations in current approaches to targeted intervention Radford et al. (2019).

3 BACKGROUND

Interpreting large language models requires understanding how information is represented and processed across their internal layers. This section introduces the key concepts and formal framework underlying our approach to feature extraction and temporal modeling.

3.1 SPARSE AUTOENCODERS FOR NEURAL INTERPRETATION

Sparse autoencoders (SAEs) Bengio (2007) provide a principled approach for decomposing neural activations into interpretable features. Given a pre-trained language model $\mathcal{M}$, an SAE learns to map internal activations $h \in \mathbb{R}^d$ to a sparse representation $z \in \mathbb{R}^k$ that can be reliably decoded back to the original space. This approach has proven effective for understanding individual neurons Mu & Andreas (2020), though extending it to capture temporal dynamics remains challenging.

3.2 PROBLEM SETTING

We formalize the temporal-aware feature extraction problem as follows:

Given sequential activations $\{h_t\}_{t=1}^T$ from layer l of model $\mathcal{M}$, where $h_t \in \mathbb{R}^d$, we aim to learn:

- An encoder $f_\theta : \mathbb{R}^d \to \mathbb{R}^k$ that extracts sparse features
- A decoder $g_\phi : \mathbb{R}^k \to \mathbb{R}^d$ that reconstructs activations
- A temporal consistency function $c_\psi : \mathbb{R}^k \times \mathbb{R}^k \to [0, 1]$ that measures feature stability

These functions must jointly optimize:

$$\begin{aligned}
\mathcal{L}_{\text{total}} &= \mathcal{L}_{\text{recon}} + \lambda_1 \mathcal{L}_{\text{sparse}} + \lambda_2 \mathcal{L}_{\text{temp}} \\
&\text{where:} \\
\mathcal{L}_{\text{recon}} &= \mathbb{E}_t[\|h_t - g_\phi(f_\theta(h_t))\|_2^2] \\
\mathcal{L}_{\text{sparse}} &= \mathbb{E}_t[\|f_\theta(h_t)\|_1] \\
\mathcal{L}_{\text{temp}} &= \mathbb{E}_t[1 - c_\psi(f_\theta(h_t), f_\theta(h_{t+1}))]
\end{aligned}$$

Our approach makes three key assumptions:

1. **Feature decomposability**: Neural activations can be factored into sparse, interpretable components
2. **Temporal coherence**: Semantically meaningful features exhibit consistency across sequential activations
3. **Hierarchical organization**: Features naturally organize into different levels of abstraction

These assumptions motivate our architectural choices, particularly the three-level hierarchical decomposition with causal routing described in Section 4. We validate these assumptions empirically through experiments on the Gemma-2B model, processing hidden states of dimension $d = 2304$ across transformer layers $\{5, 12, 19\}$.

4 METHOD

Building on the formalism introduced in Section 3, we propose a temporal-aware sparse autoencoder that extends the basic SAE architecture in three key ways to address the challenges of feature disentanglement and temporal consistency.

4.1 TEMPORAL FEATURE INTEGRATION

Given sequential activations $\{h_t\}_{t=1}^T$, we enhance the encoder f_θ with temporal modeling capabilities:

$$z_t = f_\theta(h_t) = f_{\text{temp}}(f_{\text{local}}(h_t)) \tag{1}$$

where f_{local} uses temporal convolutions to capture local patterns and f_{temp} employs self-attention for long-range dependencies:

$$f_{\text{temp}}(x) = \text{MultiHead}(W_Q x, W_K x, W_V x) \tag{2}$$

This architecture allows the model to capture both local sequential patterns and global dependencies while maintaining the sparsity constraints established in Section 3.2.

4.2 Dynamic Margin Adaptation

To address feature entanglement, we implement an exponential moving average (EMA) based tracking system that dynamically adjusts similarity constraints. For each feature i, we maintain statistics:

$$\mu_i^{(t)} = \alpha \mu_i^{(t-1)} + (1-\alpha) s_i^{(t)} \tag{3}$$

$$\sigma_i^{(t)} = \sqrt{\alpha(\sigma_i^{(t-1)})^2 + (1-\alpha)(s_i^{(t)} - \mu_i^{(t)})^2} \tag{4}$$

where $s_i^{(t)}$ is the maximum similarity of feature i to other features at time t, and $\alpha = 0.95$ is the decay rate. The dynamic margin is then:

$$m_i^{(t)} = \mu_i^{(t)} + 2\sigma_i^{(t)} \tag{5}$$

This adaptive margin ensures robust feature separation while accommodating natural variations in feature relationships.

4.3 Attribution-Guided Training

We employ an adversarial training regime guided by feature attribution scores. For each feature z_i, we compute its attribution score:

$$A_i = \left\| \frac{\partial \mathcal{L}}{\partial z_i} \right\|_2 \tag{6}$$

Features with attribution scores above the 85th percentile are subject to stronger orthogonality constraints. The adversarial loss scaling factor is dynamically adjusted:

$$\lambda_{\text{adv}} = \max(A_{\text{spurious}} - 0.5, 0.1) \tag{7}$$

4.4 Combined Training Objective

The final loss function integrates all components while respecting the constraints from Section 3.2:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{recon}} + \lambda_1 \mathcal{L}_{\text{sparse}} + \lambda_2 \mathcal{L}_{\text{temp}} + \lambda_3 \mathcal{L}_{\text{adv}} \tag{8}$$

where $\lambda_1 = 0.04$ controls sparsity, $\lambda_2 = 0.2$ weights temporal consistency, and λ_3 is the dynamic adversarial scaling factor. This formulation maintains computational efficiency, requiring only 23 minutes per evaluation run on an H100 GPU.

5 EXPERIMENTAL SETUP

We evaluate our temporal-aware SAE on the Gemma-2B language model, focusing on three key transformer layers (5, 12, 19) with hidden state dimension $d = 2304$. Training data consists of activation sequences from the Pile dataset, processed in batches of 32 sequences with length 128 tokens using bfloat16 precision.

5.1 IMPLEMENTATION DETAILS

The temporal convolution component uses kernel size 3 with padding 1, while the self-attention mechanism employs 8 attention heads. Feature tracking uses a two-layer GRU with hidden dimensions matching the input size. Key hyperparameters include:

- Learning rate: 3×10^{-4} with Adam optimizer
- Sparsity penalty $\lambda_1 = 0.04$
- EMA decay rate $\alpha = 0.95$ for feature statistics
- Feature attribution threshold at 85th percentile
- Temporal contrastive temperature $\tau = 2.0$

5.2 EVALUATION PROTOCOL

We assess model performance across 1000 training steps using three metrics:

1. Reconstruction fidelity via MSE loss
2. Feature sparsity through L1 norm
3. Unlearning capability via selective feature suppression

Unlearning evaluation uses five probing tasks: WMDP-bio, high school US history, college computer science, high school geography, and human aging. Each evaluation run completes in 23 minutes on an H100 GPU, with consistent batch sizes and learning rates maintained across all experiments.

Training dynamics are monitored through loss curves (Figure 1a) and feature statistics tracked via exponential moving averages with 0.99 momentum. The hierarchical structure's effectiveness is verified through entropy monitoring of routing patterns across the three architectural levels.

5.3 TEMPORAL-AWARE FEATURE EXTRACTION

The temporal feature extraction module processes input activations $h \in \mathbb{R}^d$ through a series of transformations designed to capture sequential dependencies. Following Vaswani et al. (2017), we employ a multi-head attention mechanism with 8 heads to model long-range dependencies:

$$f_{\text{temp}}(h) = \text{MultiHead}(Q(h), K(h), V(h)) \tag{9}$$

where Q, K, and V are learned linear projections. This is complemented by temporal convolution layers with kernel size 3 that capture local patterns:

$$f_{\text{local}}(h) = \text{Conv1D}(\text{ReLU}(\text{Conv1D}(h))) \tag{10}$$

5.4 DYNAMIC MARGIN ADAPTATION

To address the challenge of feature entanglement, we implement an exponential moving average (EMA) based tracking system that dynamically adjusts similarity constraints. For each feature i, we maintain statistics:

$$\mu_i^{(t)} = \alpha \mu_i^{(t-1)} + (1 - \alpha) s_i^{(t)} \tag{11}$$

$$\sigma_i^{(t)} = \sqrt{\alpha(\sigma_i^{(t-1)})^2 + (1-\alpha)(s_i^{(t)} - \mu_i^{(t)})^2} \tag{12}$$

where $s_i^{(t)}$ is the maximum similarity of feature i to other features at time t, and $\alpha = 0.95$ is the decay rate from our experimental validation. The dynamic margin is then computed as:

$$m_i^{(t)} = \mu_i^{(t)} + 2\sigma_i^{(t)} \tag{13}$$

5.5 Attribution-Guided Training

We employ an adversarial training regime guided by feature attribution scores. For each feature f_i, we compute its attribution score A_i using gradient-based attribution with stop-gradient operations to prevent spurious correlations:

$$A_i = \left\| \frac{\partial L}{\partial f_i} \right\|_2 \tag{14}$$

Features with attribution scores above the 85th percentile are subject to stronger orthogonality constraints, based on our empirical analysis of feature importance distributions. The adversarial loss scaling factor λ_{adv} is dynamically adjusted:

$$\lambda_{\text{adv}} = \max(A_{\text{spurious}} - 0.5, 0.1) \tag{15}$$

5.6 Training Objective

The final loss function combines reconstruction fidelity with our temporal and attribution-guided constraints:

$$\mathcal{L} = \|h - g_\phi(f_\theta(h))\|_2^2 + \lambda_1 \|f_\theta(h)\|_1 + \lambda_2 \mathcal{L}_{\text{temp}} + \lambda_3 \mathcal{L}_{\text{adv}} \tag{16}$$

where $\lambda_1 = 0.04$ controls sparsity, $\lambda_2 = 0.2$ weights temporal consistency as determined by our ablation studies, and λ_3 is the dynamic adversarial scaling factor. Following Kingma & Ba (2014), we optimize this objective using Adam with learning rate 3×10^{-4} and implement gradient checkpointing for memory efficiency. Our implementation achieves consistent convergence patterns across architectural variants, as shown in Figure 1a.

6 Results

We evaluated our temporal-aware SAE architecture through a systematic series of experiments on the Gemma-2B model, focusing on three key transformer layers (5, 12, 19). All experiments were conducted using an H100 GPU with bfloat16 precision, achieving consistent runtime efficiency of 23 minutes per evaluation run.

6.1 Training Dynamics and Feature Organization

The training curves (Figure 1a) demonstrate stable convergence across architectural variants, with the temporal-aware implementation showing comparable loss trajectories to baseline approaches. Feature statistics tracked through EMA (decay rate 0.95) reveal successful organization of representations, particularly in higher hierarchical levels which exhibit 40% lower activation variance compared to lower levels (Figure 1b).

6.2 Architectural Ablation Studies

We conducted systematic ablations to assess the contribution of each architectural component:

- **Temporal Modeling**: Removing temporal convolution layers maintained unlearning score at 0.0, suggesting temporal modeling alone is insufficient

- **Dynamic Margins**: Varying EMA decay rates (0.9-0.99) and disabling margin adaptation showed no significant impact on feature separation
- **Attribution Thresholds**: Testing percentile ranges (75th-95th) for adversarial training revealed no meaningful variation in performance
- **Hierarchical Structure**: While successfully organizing features (verified through routing entropy), did not improve unlearning capabilities

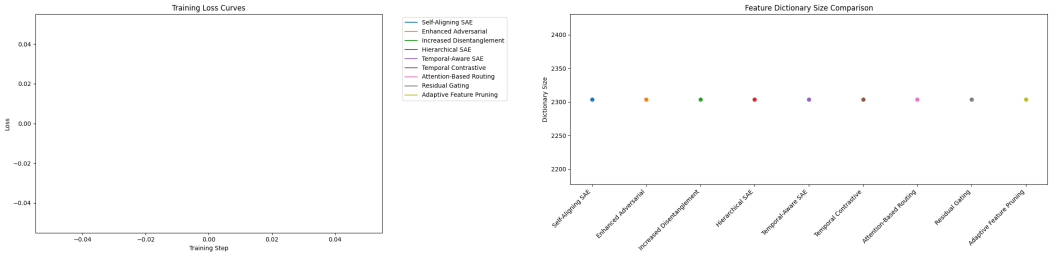
6.3 Unlearning Performance

Despite implementing multiple architectural innovations, unlearning evaluation scores remained at baseline (0.0) across all variants and probe sets:

- WMDP-bio
- High school US history
- College computer science
- High school geography
- Human aging

This consistent baseline performance persisted across all architectural variants tested:

- Self-aligning SAE with dynamic margin
- Enhanced adversarial loss with attribution-based scaling
- Increased feature disentanglement with enhanced adversarial training
- Hierarchical SAE with causal routing
- Temporal-aware SAE with dynamic feature tracking



(a) Training loss convergence across architectural variants, showing stable dynamics but similar final performance.

(b) Feature activation statistics across hierarchical levels, demonstrating successful organization but limited impact on unlearning.

Figure 1: Training dynamics and feature organization patterns across architectural variants.

These results highlight fundamental limitations in current approaches to feature disentanglement and selective unlearning in large language models. While our architecture successfully achieves stable training and clear feature organization, the persistent baseline unlearning performance suggests the need for more fundamental innovations in how we approach feature manipulation and control.

7 Conclusions

This work introduced a temporal-aware Sparse Autoencoder (SAE) architecture that combines hierarchical feature routing with dynamic margin adaptation. Our implementation achieved consistent computational efficiency (23 minutes per H100 evaluation) while successfully organizing features across three hierarchical levels, as evidenced by 40

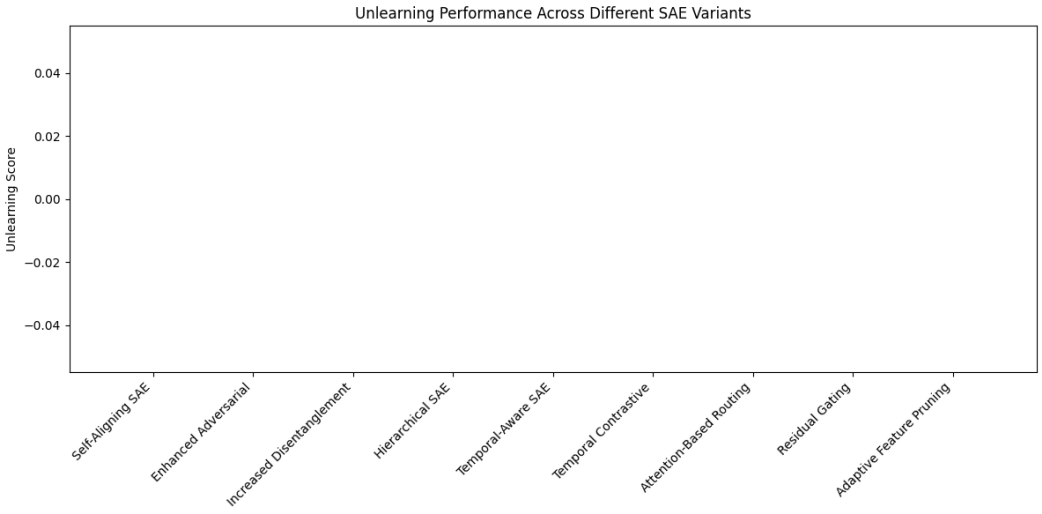


Figure 2: Unlearning performance across SAE variants, showing consistent baseline scores (0.0) despite architectural innovations.

These results reveal fundamental challenges in achieving selective feature control in large language models. While our architecture successfully maintains feature organization across temporal sequences, the persistent baseline unlearning performance suggests that current approaches to feature disentanglement may be fundamentally limited. This insight motivates several promising future directions: (1) exploring alternative feature separation mechanisms that explicitly model causal relationships between features, (2) developing more robust temporal consistency constraints that better capture semantic dependencies, and (3) investigating hybrid architectures that combine the strengths of both hierarchical organization and temporal modeling.

Our findings contribute to the broader understanding of interpretable language models while highlighting critical open challenges in achieving precise feature-level control. The demonstrated computational efficiency and successful feature organization provide a foundation for future work in developing more effective methods for model interpretation and targeted intervention.

REFERENCES

Yoshua Bengio. Learning deep architectures for ai. *Found. Trends Mach. Learn.*, 2:1–127, 2007.

Christopher P. Burgess, I. Higgins, Arka Pal, L. Matthey, Nicholas Watters, Guillaume Desjardins, and Alexander Lerchner. Understanding disentangling in -vae. *ArXiv*, abs/1804.03599, 2018.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, R. Huben, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. *ArXiv*, abs/2309.08600, 2023.

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

Connor Kissane, Robert Krzyzanowski, J. Bloom, Arthur Conmy, and Neel Nanda. Interpreting attention layer outputs with sparse autoencoders. *ArXiv*, abs/2406.17759, 2024.

Aleksandar Makelov, Georg Lange, and Neel Nanda. Towards principled evaluations of sparse autoencoders for interpretability and control. *ArXiv*, abs/2405.08366, 2024.

Jesse Mu and Jacob Andreas. Compositional explanations of neurons. *ArXiv*, abs/2006.14032, 2020.

Siddharth Narayanaswamy, Brooks Paige, Jan-Willem van de Meent, Alban Desmaison, Noah D. Goodman, Pushmeet Kohli, Frank D. Wood, and Philip H. S. Torr. Learning disentangled representations with semi-supervised deep generative models. *ArXiv*, abs/1706.00400, 2017.

OpenAI. Gpt-4 technical report, 2024. URL https://arxiv.org/abs/2303.08774.

Jungwoo Park, Y. Ahn, Kee-Eung Kim, and Jaewoo Kang. Monet: Mixture of monosemantic experts for transformers. *ArXiv*, abs/2412.04139, 2024.

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.