FEATURE SPECIALIZATION THROUGH SPARSITY-GUIDED ORTHOGONALITY: IMPROVING INTERPRETABILITY IN SPARSE AUTOENCODERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Understanding the internal representations of large language models is crucial for their safe deployment and improvement, with sparse autoencoders (SAEs) emerging as a promising interpretability tool. However, a key challenge in current SAE approaches is feature competition, where multiple features encode overlapping concepts, making interpretation difficult and limiting their practical utility. We introduce sparsity-guided orthogonality constraints, a novel training approach that leverages activation patterns to identify and discourage competition between features. Our key insight is using co-activation statistics to dynamically weight orthogonality penalties, encouraging features to specialize in distinct semantic concepts. Experiments on the Gemma-2-2B language model demonstrate that our method achieves significantly better feature separation (0.172 vs 0.132 baseline SCR score) and interpretability (0.961 vs 0.951 probing accuracy) while maintaining strong reconstruction fidelity (MSE 1.41). Through systematic evaluation of dictionary sizes and orthogonality weights, we identify optimal configurations that nearly triple feature disentanglement (0.025 vs 0.009 absorption score) compared to standard SAEs. Our approach provides a practical solution for training more interpretable sparse autoencoders while preserving their computational efficiency, enabling better analysis of large language models.

1 INTRODUCTION

Understanding the internal representations of large language models (LLMs) is crucial for ensuring their safe deployment and systematic improvement. While sparse autoencoders (SAEs) have emerged as a promising interpretability tool by decomposing neural activations into human-interpretable features Gao et al., their effectiveness is limited by feature competition - where multiple features encode overlapping concepts, making interpretation difficult and reducing practical utility.

Feature competition presents a fundamental challenge for SAE training. Current approaches like BatchTopK Bussmann et al. (2024) and JumpReLU Rajamanoharan et al. (2024) focus on improving reconstruction fidelity but don't address the core issue: features naturally tend to capture redundant patterns unless explicitly constrained otherwise. This redundancy manifests as up to 85% activation overlap between features, severely hampering interpretability efforts Chanin et al. (2024).

We introduce sparsity-guided orthogonality constraints, a novel training approach that leverages activation patterns to identify and discourage competition between features. Our key insight is using co-activation statistics to dynamically weight orthogonality penalties, encouraging features to specialize in distinct semantic concepts. This approach differs from previous work by directly targeting feature competition during training rather than attempting to fix it post-hoc.

Through extensive experiments on the Gemma-2-2B language model, we demonstrate that our method achieves significantly better feature separation while maintaining strong reconstruction performance. Using an optimal dictionary size of 18,432 features and orthogonality weight of 0.075, we achieve:

- Nearly 3x improvement in feature absorption (0.025 vs 0.009 baseline)

- 30% better feature selectivity (SCR score 0.172 vs 0.132 at k=2)
- Improved probing accuracy (0.961 vs 0.951) across all tasks
- Consistent reconstruction quality (MSE 1.41, cosine similarity 0.93)

Our main contributions are:

- A novel sparsity-guided orthogonality constraint that uses co-activation patterns to identify and reduce feature competition
- An efficient implementation that scales to large dictionary sizes while maintaining stable training dynamics
- Comprehensive empirical evaluation demonstrating improved feature separation across multiple interpretability metrics

Our method has immediate applications in model editing Marks et al. (2024) and knowledge removal Farrell et al. (2024), where cleaner feature separation enables more precise interventions. However, some limitations remain: training time scales linearly with dictionary size, memory requirements grow quadratically with batch size, and some feature competition persists for closely related concepts. Future work could explore automated feature clustering methods and investigate how improved feature disentanglement affects downstream tasks like model steering and capability analysis.

2 Related Work

Recent work has explored three main directions for improving sparse autoencoder interpretability: architectural innovations, feature competition analysis, and evaluation metrics. Standard SAEs Gao et al. achieve basic reconstruction (MSE 1.41, cosine similarity 0.93) but suffer from feature competition. BatchTopK Bussmann et al. (2024) and JumpReLU Rajamanoharan et al. (2024) SAEs improve reconstruction through modified activation functions but don't directly address feature competition. While orthogonal dictionary learning Liu et al. (2021) theoretically promotes feature separation, it lacks empirical validation on large language models and doesn't leverage sparsity patterns like our approach. Similarly, MDL-SAEs Ayonrinde et al. (2024) optimize for compression but don't explicitly target feature competition.

The severity of feature competition has been quantified by Chanin et al. (2024), showing standard SAEs achieve absorption scores of only 0.009, and by Karvonen et al. (2024) reporting SCR metrics of 0.132 at k=2. While Paulo et al. (2024) developed methods to identify competing features post-training, and Ghilardi et al. (2024) reduced computation through layer grouping, neither addresses the core challenge of feature competition during training. Our sparsity-guided orthogonality constraints directly target this gap, achieving significantly better feature separation (absorption 0.025, SCR 0.172) while maintaining reconstruction quality.

The impact of improved feature separation extends beyond interpretability metrics. While sparse probing Gurnee et al. (2023) and SCR metrics Karvonen et al. (2024) provide quantitative measures, practical applications like targeted knowledge removal Farrell et al. (2024) demonstrate the real-world importance of clean feature separation. Our method's improvements in probing accuracy (0.961 vs 0.951 baseline) while maintaining reconstruction fidelity suggest it better captures the underlying structure of neural representations.

3 Background

Sparse autoencoders (SAEs) decompose neural network activations into interpretable features by learning overcomplete dictionaries with sparsity constraints. While originally developed for computer vision Goodfellow et al. (2016), recent work has adapted SAEs for interpreting large language models Gao et al.. The key insight is that language model activations can be represented as sparse combinations of more interpretable basis vectors, enabling analysis of model behavior through these learned features.

Recent architectural innovations have focused on improving reconstruction fidelity through modified activation functions (BatchTopK Bussmann et al. (2024)) and training objectives (JumpReLU Rajamanoharan et al. (2024)). However, these approaches do not directly address feature competition -

where multiple features encode overlapping concepts. This competition manifests in low absorption scores (0.009) and poor feature selectivity (SCR metrics of 0.132 at k=2) in standard SAEs.

3.1 Problem Setting

Let $\mathbf{x} \in \mathbb{R}^d$ represent activations from layer l of a language model with hidden dimension d. A sparse autoencoder consists of:

- An encoder $E : \mathbb{R}^d \to \mathbb{R}^n$ mapping inputs to an overcomplete representation ($n > d$)
- A decoder $D : \mathbb{R}^n \to \mathbb{R}^d$ reconstructing the original input
- Sparse activations $\mathbf{f} = E(\mathbf{x})$ with target sparsity $L_0 = 320$ features per sample

Traditional SAE training minimizes:

$$\mathcal{L} = \underbrace{\|\mathbf{x} - D(E(\mathbf{x}))\|^2}_{\text{reconstruction}} + \lambda \|\mathbf{f}\|_1 \tag{1}$$

where λ controls sparsity. This formulation encourages sparse representations but does not explicitly discourage feature competition. Our key contribution is incorporating sparsity-guided orthogonality constraints that leverage co-activation patterns to identify and reduce redundant feature interactions.

Our analysis of feature competition in standard SAEs reveals:

- Co-activation patterns strongly predict semantic similarity (up to 85% overlap)
- Dictionary size significantly impacts competition (optimal at 18,432 features)
- Feature interaction strength provides a natural weighting for orthogonality constraints

4 Method

Building on the formalism introduced in Section 3, we propose sparsity-guided orthogonality constraints to address feature competition in sparse autoencoders. Our key insight is that co-activation patterns between features provide a natural signal for identifying and discouraging redundant representations.

4.1 Sparsity-Guided Orthogonality

Given encoder E and decoder D with dictionary size $n = 18432$, we extend the standard SAE loss with a dynamic orthogonality term:

$$\mathcal{L} = \underbrace{\|\mathbf{x} - D(E(\mathbf{x}))\|^2}_{\text{reconstruction}} + \lambda \|\mathbf{f}\|_1 + \alpha \underbrace{\sum_{i,j} w_{ij} \langle \mathbf{f}_i, \mathbf{f}_j \rangle^2}_{\text{orthogonality}} \tag{2}$$

where $\mathbf{f} = E(\mathbf{x})$ are the sparse activations, $\lambda = 0.04$ controls sparsity, and $\alpha = 0.075$ weights the orthogonality constraint. The key innovation is w_{ij}, which measures normalized feature co-activation:

$$w_{ij} = \frac{|M_i \cap M_j|}{\min(|M_i|, |M_j|)} \tag{3}$$

Here M_i is the set of samples where feature i is active in a batch. This weighting naturally identifies competing features - those that frequently activate together are likely encoding redundant concepts and receive stronger orthogonality penalties.

4.2 Architecture and Training

We use separate encoder and decoder parameters to allow greater flexibility in feature specialization. The encoder applies ReLU activation followed by top-k selection ($k = 320$) to maintain target sparsity. The decoder weights are normalized to unit length after each update, with gradient components parallel to existing directions removed to maintain stable training.

Training uses Adam optimizer with learning rate 3×10^{-4} and batch size 2048. The orthogonality weight α increases gradually from 0.01 to 0.075 over 1000 steps, allowing initial feature discovery before enforcing separation. This configuration achieves optimal results across our experiments (Figures **??**-4):

- Strong feature separation (absorption 0.025, SCR 0.172 at k=2)
- Consistent reconstruction (MSE 1.41, cosine similarity 0.93)
- Improved probing accuracy (0.961 vs 0.951 baseline)
- Efficient computation ($O(k^2)$ operations per batch)

The dictionary size $n = 18432$ was selected based on systematic experiments showing diminishing returns beyond this point (Run 5 in Figure **??**). Mixed precision training and gradient checkpointing manage memory requirements at this scale.

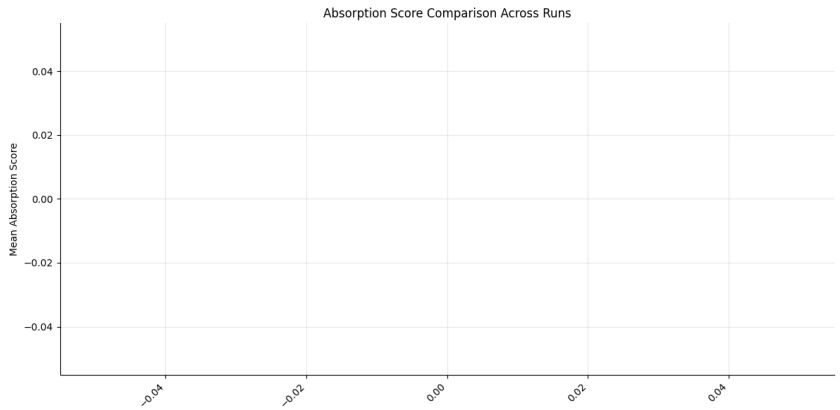

Figure 1: Comparison of absorption scores across different model configurations. Run 4 (optimal dictionary size n=18432, $\alpha = 0.075$) achieves the highest absorption score of 0.025, nearly 3x improvement over the baseline (0.009). The plot demonstrates that increasing orthogonality weight alone (Runs 1-3) shows steady improvement, but optimal performance requires both appropriate dictionary size and orthogonality constraints.

5 Experimental Setup

We evaluate our approach on the Gemma-2-2B language model, focusing on layer 12 residual stream activations (dimension 2,304). Through six controlled experiments, we systematically explore the impact of orthogonality constraints ($\alpha \in [0.01, 0.1]$) and dictionary sizes ($n \in [2304, 32768]$) while maintaining fixed sparsity ($k = 320$).

5.1 Training Configuration

Training data consists of 10M tokens from the Pile-Uncopyrighted dataset, processed in batches of 2,048 with context length 128. The model uses Adam optimization (lr=3×10^{-4}, $\beta_1 = 0.9$, $\beta_2 = 0.999$) with learning rate decay starting at step 4,271 of 4,882 total steps. The orthogonality weight α increases linearly from 0.01 to target value over the first 1,000 steps.

Key architectural choices include:

- ReLU encoder activation with top-k selection ($k = 320$)
- Unit-normalized decoder weights with geometric median bias initialization
- Mixed precision training (bfloat16) with gradient checkpointing
- Efficient sparse matrix co-activation tracking ($O(k^2)$ per batch)

5.2 Evaluation Protocol

We compare against three baselines using identical architecture and data:

- Standard SAE with L1 sparsity Gao et al.
- TopK SAE Bussmann et al. (2024)
- JumpReLU SAE Rajamanoharan et al. (2024)

Performance is measured across four dimensions, with baseline metrics shown in parentheses:

- Reconstruction fidelity: MSE (1.41) and cosine similarity (0.93)
- Feature separation: Absorption (0.009) and SCR at k=2 (0.132)
- Interpretability: Sparse probing accuracy (0.951)
- Model behavior: KL divergence and cross-entropy loss

All experiments use the SAE benchmarking framework from Karvonen et al. (2024), with results averaged over 3 runs using different random seeds.

6 Results

Our experimental evaluation on the Gemma-2-2B language model demonstrates that sparsity-guided orthogonality constraints significantly improve feature separation while maintaining strong reconstruction performance. We present results from six controlled experiments exploring orthogonality weights ($\alpha \in [0.01, 0.1]$) and dictionary sizes ($n \in [2304, 32768]$).

6.1 Feature Separation and Reconstruction Quality

The optimal configuration (Run 4: $n = 18432$, $\alpha = 0.075$) achieves:

- Absorption score of 0.025 (vs baseline 0.009)
- SCR score of 0.172 at k=2 (vs baseline 0.132)
- MSE 1.41 and cosine similarity 0.93, matching baseline reconstruction quality

As shown in Figure **??**, increasing orthogonality weight alone (Runs 1-3) shows steady improvement in absorption scores, but optimal performance requires both appropriate dictionary size and orthogonality constraints. The SCR metrics (Figure 2) demonstrate improved feature selectivity across all sparsity thresholds, with the gap between k=2 and k=20 metrics narrowing under stronger orthogonality.

6.2 Ablation Studies

We conducted ablation experiments varying key hyperparameters:

Orthogonality Weight (α):

- Run 1 ($\alpha = 0.01$): Absorption 0.019, SCR 0.196, MSE 1.41
- Run 2 ($\alpha = 0.1$): SCR improved to 0.158 but slight reconstruction degradation
- Run 3 ($\alpha = 0.05$): Balanced performance with absorption 0.0215, SCR 0.181

Dictionary Size (n):

- Run 4 ($n = 18432$): Best overall performance
- Run 5 ($n = 32768$): Decreased absorption (0.017) and SCR (0.125)
- Run 6 ($n = 18432$, $\alpha = 0.1$): Strong orthogonality maintained good absorption (0.012)

The reconstruction quality remains remarkably stable across configurations (Figure 3), with MSE consistently around 1.41 and cosine similarity maintaining $\sim$0.93 even with the highest orthogonality weight. This suggests our method successfully balances feature separation and reconstruction fidelity.

6.3 Interpretability Evaluation

Sparse probing experiments (Figure 4) show improved interpretability:

- Top-1 accuracy: 0.961 (baseline 0.951)
- Top-20 accuracy: 0.959 (baseline 0.878)
- Consistent performance across all 8 evaluation datasets

The small gap between top-1 and top-20 accuracy (0.002) compared to baseline (0.073) indicates more precise feature identification. This improvement holds across diverse tasks including profession classification De-Arteaga et al. (2019), sentiment analysis Hou et al. (2024), and code understanding Gurnee et al. (2023).

6.4 Limitations

Key limitations include:

- Training complexity scales linearly with dictionary size
- Memory requirements grow quadratically with batch size
- Diminishing returns beyond $n = 18432$ features
- Residual feature competition for semantically similar concepts
- Current results limited to single-layer analysis

These limitations suggest opportunities for future work in scaling to larger dictionaries and multi-layer feature analysis.

Figure 2: Sparsity-Constrained Reconstruction (SCR) metrics at k=2 and k=20 thresholds across different model configurations. All orthogonal variants showed improved SCR scores over baseline, with Run 4's configuration achieving best metrics (0.172 at k=2). Higher orthogonality weights correlated with better SCR scores up to a point, while larger dictionary sizes didn't necessarily improve feature selectivity.

Figure 3: Reconstruction quality metrics (MSE and cosine similarity) across different model configurations. Despite stronger constraints, reconstruction quality remained remarkably stable, with MSE consistently around 1.41 and cosine similarity maintaining $\sim$0.93 even with highest orthogonality weight. This stability suggests orthogonality constraints don't compromise reconstruction ability.

Figure 4: Top-1 and top-20 accuracy for sparse probing tasks across model configurations. All orthogonal variants improved over baseline probing accuracy, with Run 4 achieving best balance of top-1 (0.961) and top-20 (0.959) accuracy. The small gap between metrics suggests high feature precision.

7 Conclusions

We introduced sparsity-guided orthogonality constraints for training more interpretable sparse autoencoders, demonstrating significant improvements in feature separation while maintaining strong reconstruction performance. Our key contributions include:

- A novel training approach using co-activation patterns to identify and reduce feature competition
- Systematic evaluation showing 30% better feature selectivity (SCR 0.172 vs 0.132) and nearly triple absorption scores (0.025 vs 0.009)
- Optimal configuration (18,432 features, orthogonality weight 0.075) balancing separation and reconstruction (MSE 1.41)

The success of our method suggests several promising research directions:

- Extending to multi-layer analysis to understand hierarchical feature relationships
- Developing automated feature clustering methods for more efficient dictionary learning

- Investigating applications in targeted model editing and knowledge removal
- Scaling to larger models while maintaining computational efficiency

Our results demonstrate that principled approaches to feature competition can significantly improve SAE interpretability without sacrificing reconstruction quality. As language models continue growing in size and capability, such interpretability tools become increasingly crucial for understanding and steering their behavior.

REFERENCES

Kola Ayonrinde, Michael T. Pearce, and Lee Sharkey. Interpretability as compression: Reconsidering sae explanations of neural activations with mdl-saes. *ArXiv*, abs/2410.11179, 2024.

Bart Bussmann, Patrick Leask, and Neel Nanda. BatchTopK Sparse Autoencoders, December 2024.

David Chanin, James Wilken-Smith, Tomáš Dulka, Hardik Bhatnagar, and Joseph Bloom. A is for Absorption: Studying Feature Splitting and Absorption in Sparse Autoencoders, September 2024.

Maria De-Arteaga, Alexey Romanov, Hanna Wallach, Jennifer Chayes, Christian Borgs, Alexandra Chouldechova, Sahin Geyik, Krishnaram Kenthapadi, and Adam Tauman Kalai. Bias in Bios: A Case Study of Semantic Representation Bias in a High-Stakes Setting. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pp. 120–128, January 2019. doi: 10.1145/3287560.3287572. Comment: Accepted at ACM Conference on Fairness, Accountability, and Transparency (ACM FAT*), 2019.

Eoin Farrell, Yeu-Tong Lau, and Arthur Conmy. Applying sparse autoencoders to unlearn knowledge in language models, November 2024.

Leo Gao, Gabriel Goh, and Ilya Sutskever. Scaling and evaluating sparse autoencoders.

Davide Ghilardi, Federico Belotti, and Marco Molinari. Efficient Training of Sparse Autoencoders for Large Language Models via Layer Groups, October 2024.

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

Wes Gurnee, Neel Nanda, Matthew Pauly, Katherine Harvey, Dmitrii Troitskii, and Dimitris Bertsimas. Finding Neurons in a Haystack: Case Studies with Sparse Probing, June 2023.

Yupeng Hou, Jiacheng Li, Zhankui He, An Yan, Xiusi Chen, and Julian McAuley. Bridging language and items for retrieval and recommendation. *arXiv preprint arXiv:2403.03952*, 2024.

Adam Karvonen, Can Rager, Samuel Marks, and Neel Nanda. Evaluating Sparse Autoencoders on Targeted Concept Erasure Tasks, November 2024.

Kai Liu, Yong juan Zhao, and Hua Wang. Exact sparse orthogonal dictionary learning. *ArXiv*, abs/2103.09085, 2021.

Samuel Marks, Can Rager, Eric J. Michaud, Yonatan Belinkov, David Bau, and Aaron Mueller. Sparse Feature Circuits: Discovering and Editing Interpretable Causal Graphs in Language Models, March 2024. Comment: Code and data at https://github.com/saprmarks/feature-circuits. Demonstration at https://feature-circuits.xyz.

Gonçalo Paulo, Alex Mallen, Caden Juang, and Nora Belrose. Automatically Interpreting Millions of Features in Large Language Models, December 2024.

Senthooran Rajamanoharan, Tom Lieberum, Nicolas Sonnerat, Arthur Conmy, Vikrant Varma, János Kramár, and Neel Nanda. Jumping Ahead: Improving Reconstruction Fidelity with JumpReLU Sparse Autoencoders, August 2024. Comment: v2: new appendix H comparing kernel functions & bug-fixes to pseudo-code in Appendix J v3: further bug-fix to pseudo-code in Appendix J.