Feature Integration Spaces: Joint Training Reveals Dual Encoding in Neural Network Representations

Anonymous Submission

Anonymous affiliation

Abstract

Current sparse autoencoder (SAE) approaches to neural network interpretability assume that activations can be decomposed through linear superposition into sparse, interpretable features. Despite high reconstruction fidelity, SAEs consistently fail to eliminate polysemanticity and exhibit pathological behavioral errors. We propose that neural networks encode information in two complementary spaces compressed into the same substrate: feature identity and feature integration.

To test this dual encoding hypothesis, we develop sequential and joint-training architectures to capture identity and integration patterns simultaneously. Joint training achieves 41.3% reconstruction improvement and 51.6% reduction in KL divergence errors. This architecture spontaneously develops bimodal feature organization: orthogonal features contributing to integration pathways and the rest contributing directly to the residual. Small nonlinear components (3% of parameters) achieve 16.5% standalone improvements, demonstrating parameter-efficient capture of computational relationships crucial for behavior. Additionally, intervention experiments using 2×2 factorial stimulus designs demonstrated that integration features exhibit selective sensitivity to experimental manipulations and produce systematic behavioral effects on model outputs, including significant interaction effects across semantic dimensions.

This work provides systematic evidence for (1) dual-encoding in neural representations, (2) meaningful nonlinearly encoded feature interactions, and (3) introduces an architectural paradigm shift from post-hoc feature analysis to integrated computational design, establishing foundations for next-generation SAEs.

Code — Available upon request
Datasets — WikiText-103 (publicly available)
Extended version — Available upon request

Introduction

The Linear Superposition Assumption

Current interpretability approaches fundamentally assume that neural network representations follow a linear superposition model (Elhage et al. 2022), where each activation can be decomposed into a sparse combination of interpretable features:

$$neural\_activation = w_1 \times feature_1 + w_2 \times feature_2 + \ldots$$

This assumption underlies the success of Sparse Autoencoders (SAEs) (Bricken et al. 2023), which have demonstrated remarkable ability to discover interpretable features and achieve high reconstruction fidelity on neural activations. Within this framework, non-orthogonal feature representations are viewed as interference or compression artifacts—necessary evils that arise when neural networks attempt to represent more features than they have dimensions.

However, a fundamental puzzle remains: despite achieving high reconstruction fidelity, SAEs consistently fail to eliminate polysemantic features that respond to seemingly unrelated concepts (Bricken et al. 2023; Cunningham et al. 2023; Templeton et al. 2024; Chen et al. 2024) and exhibit pathological behavioral errors when their reconstructions replace original activations[1]. If linear superposition fully captures neural computation, why do the same polysemantic patterns appear robustly across different models and scales? This persistence suggests that our current understanding may be incomplete.

Recent work has highlighted systematic limitations in sparse coding approaches[1,2]. Gurnee et al. demonstrated that SAE reconstruction errors are pathological rather than random, indicating missing computational structure.[1] The mechanistic interpretability literature describes substantial "dark matter"[2]—neural computation that remains unexplained even after extensive circuit analysis (Olah et al. 2020). These findings collectively suggest that the linear superposition assumption may be insufficient to capture the full complexity of neural representations.

Dual Encoding Hypothesis

We propose that neural networks encode information in two complementary spaces that are compressed into the same neural substrate:

- **Feature Identity Space**: Represents what concepts are present in the input. This corresponds to the sparse features successfully captured by current SAE approaches—interpretable concepts like "Paris," "democracy," or "positive sentiment" that can be identified and measured independently.
- **Feature Integration Space**: Represents how concepts combine computationally to produce emergent meanings and behaviors. This may include the relationships between features that cannot be captured by linear combinations—the computational patterns that determine how "surprise" + "birthday" produces joy while "surprise" + "diagnosis" produces anxiety.

This dual encoding framework reframes non-orthogonal representations not as interference to be eliminated, but as computational structure encoding meaningful relationships between concepts. The persistent polysemanticity observed in neural networks may reflect the compression of both identity and integration information into the same representational space, rather than mere artifacts of insufficient capacity.

Additionally, this dual encoding hypothesis is distinct from existing analyses of feature relationships through co-activation patterns or similarity metrics, which capture statistical correlations between features across datasets. Instead, we focus on computational interactions—how features combine to produce emergent meanings that cannot be predicted from their individual activation patterns or co-occurrence statistics. While similarity analysis might reveal that "fire" and "hearth" (or "fire" and "forest") features often appear together, integration analysis reveals how their combination computes concepts like "warmth/comfort" (or "destruction/emergency") with behavioral consequences that emerge only from their joint activation.

Finally, traditional approaches attempt to address these limitations through post-hoc analysis—training SAEs first, then analyzing their failures. We propose an integrated architectural approach that captures both identity and integration patterns during training, preventing the systematic errors rather than detecting them after they occur.

Neural Compression and Computational Structure

Under this view, neural networks face a fundamental compression challenge: they must encode both the identity of relevant features and the computational relationships between them within limited representational capacity. A neuron that responds to both "late" and "party" (or "late" and "meeting") concepts may not simply be storing two unrelated features due to capacity constraints—it may be computing something about their relationship, such as "fashionable" (or "problematic").

This perspective offers a unified explanation for several puzzling phenomena in neural network interpretability: why polysemantic neurons are so robust across models, why certain feature combinations consistently appear together, and why interventions on individual features often produce complex, context-dependent effects, and why features exhibit sharp phase transitions during training as they crystallize from integration patterns into dedicated representations. If neurons encode computational relationships alongside feature identities, these observations become natural consequences of the underlying representational structure rather than obstacles to overcome.

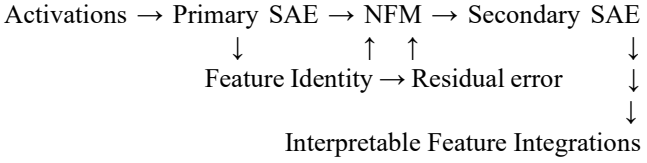
To evaluate this dual encoding hypothesis, we develop both sequential and joint training architectures that explicitly model feature identity and integration spaces. We investigate whether the improves reconstruction error, improves the pathology of logit probability distributions, whether they naturally organize into specialized computational roles when architectural constraints allow for separate feature identity and feature integration encoding, and whether non-linear feature interactions can be confirmed with behavioral experimentation.

Methods

Experimental Pipeline

Our methodology decomposes neural representations into two complementary encoding spaces through a three-stage pipeline:

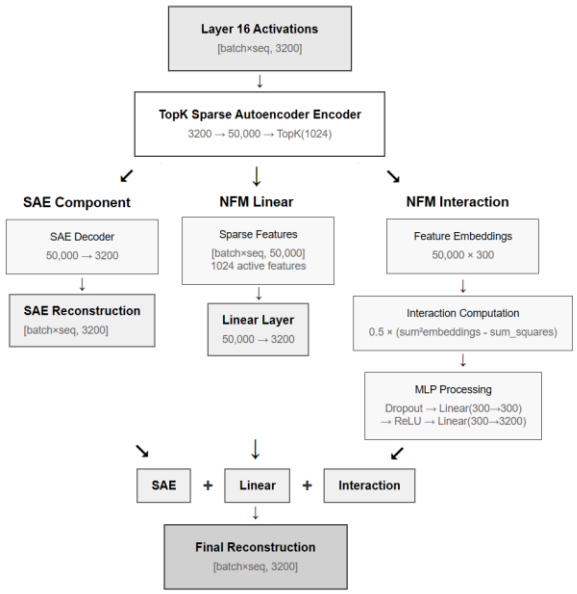
1. sparse feature extraction via SAE training,
2. integration pattern capture using Neural Factorization Machine (NFM) trained on SAE reconstruction residuals, and
3. integration space analysis through a secondary SAE decomposition of the dense NFM embeddings, as a post-hoc interpretative step



Joint Training Architecture: Following our initial exploration above, we also developed an integrated architecture that trains SAE and interactive components simultaneously in a single optimization phase, allowing natural specialization of feature types during learning.

All components were trained jointly using Adam optimizer (*lr=0.0001,* β_1*=0.9,* β_2*=0.999*) with linear learning rate decay from 1×10^{-4} to 1×10^{-5} over 80% of training steps using 5 million tokens from WikiText-103. Training proceeded in chunks of 10,000 tokens each, with 90/10 train-validation splits and batch size of 64 sequences. The TopK constraint (*1024, ~2.05% sparsity of 50k features*) provided

automatic sparsity regularization, eliminating the need for additional sparsity loss terms beyond the reconstruction objective.



Model and Data Configuration

We conducted experiments using OpenLLaMA-3B with activations extracted from layer 16 (middle layer). The model was evaluated on WikiText-103, with tokenized sequences processed in 50-token windows. All experiments utilized a single NVIDIA RTX 3090 GPU with 24GB VRAM and 128GB system RAM.

Stage 1: Sparse Autoencoder Training

We trained a 50,000-feature TopK SAE achieving 0.136 reconstruction loss and 86.4% variance explained.

Stage 2: Neural Factorization Machine Architecture

NFMs capture feature integration patterns by predicting SAE reconstruction residuals:

$$SAE\ error = x\_original - SAE(x\_original)$$
$$Residual\ Prediction\ (SAE\ error) = NFM(SAE\ features)$$

Architecture: Neural Factorization Machine with linear + interaction components

$$Linear\ Output = \Sigma_i\, w_i \times f_i + b$$
$$Interaction\ Output = 0.5 \times (\Sigma_i\, v_i f_i)^2 - \Sigma_i (v_i f_i)^2$$

where $v_i \in \mathbb{R}^k$ represents learned embedding vectors for feature i

The NFM was trained on 5 million tokens using Adam optimization (lr=1e-4) with K=300 embedding dimensions. This achieved 23.4% error reduction over the base SAE, with linear components contributing 95.5% and interactions contributing 4.5% of the improvement.

Stage 3: Integration Space Analysis

To analyze the computational structure captured by NFMs, we applied secondary TopK SAEs to the NFM interaction pathway, specifically targeting post-MLP1 vectors before ReLU activation. The secondary SAE used 25× expansion (300 → 7,500 features) with K=250 top active features of the primary SAE for each sample.

Validation methodology: We implemented 2×2 factorial stimulus designs (formal/informal × emotional/neutral) with systematic intervention experiments. Secondary SAE features were ranked by activation variance across experimental conditions, then subjected to clamping interventions at multiple levels (0×, 1×, ±4×). Behavioral effects were measured through logit changes for category-relevant vocabulary sets.

Stage 4: Experimental Validation Intervention testing:

We validated integration features through systematic clamping of both primary SAE features (via linear weight manipulation) and secondary SAE features (direct activation clamping). Effects were measured using logit differential analysis (ANOVA) across formality and emotion vocabulary categories to demonstrate specificity of the nonlinear behaviorial interaction effects of the interaction features (secondary SAE).

Controls: A secondary SAE was trained directly on the original residuals showing no added significant reconstruction loss, compared to the NFM modelling approach. Other non-dead, but less active interaction features were interrogated showing no interaction behavioral effects. As expected, linear-component-only variants also did not demonstrate nonlinear behaviorial interactions.

Stage 5: Joint Training Implementation

We trained a singular architecture which jointly trains SAE components and NFM interactions in a single loss function. The architecture combines three components in a residual manner:

$$final\_recon = sae\_reconstruction + nfm\_linear\_out + nfm\_interaction\_out$$
$$total\_loss = MSE(final\_recon - batch)$$

This approach allows features to specialize naturally for either identity representation or integration computation during training, rather than retrofitting integration capture to pre-trained SAE features. For this particular implementation, we used TopK to provide automatic sparsity regularization, without any additional explicit loss terms, and relied on MSE.

Stage 6: Evaluation of Joint Architecture

In addition to reconstruction loss, and component contributions, we also looked at:

Logit distribution: Analyzed predicted logits through KL divergence and cross-entropy loss by replacing model activations with architecture reconstructions and computing divergence from original model outputs, following Gurnee et al. methodology.

Feature Orthogonality: We analyzed feature orthogonality through Gram matrix analysis (computing pairwise dot products between feature weight vectors to assess orthogonality patterns) and PCA analysis of feature weight distributions.

Bimodal Investigation: We analyzed features with different orthogonality to investigate their differential contributions to reconstruction vs integration pathways.

Parameter Impact*:* Finally, we looked at a component-wise analysis measuring reconstruction improvements and KL divergence relative to parameter allocation across SAE, NFM linear, and NFM interaction components, relative to the parameter count and total weight.

Results

Preliminary Quantitative Reconstruction Exploration

The Neural Factorization Machine (NFM) approach achieved substantial improvements over sparse autoencoder baselines. Training on 5 million tokens, the combined SAE+NFM system demonstrated 23.18% error reduction on training data and 23.43% error reduction on validation data compared to SAE-only reconstruction which constrained our top K features to the top 250 features (*train error: 0.3813 → 0.2930*; *validation error: 0.3672 → 0.2811, [SAE only → SAE + NFM]*).

Component analysis revealed that linear combinations dominated the improvement, contributing 95.5% of the correction magnitude (linear: 0.2773, interaction: 0.0130), while higher-order (non-linear) interaction effects accounted for 4.5%. This suggests that NFMs capture both underspecified feature combinations that could be learnable by much larger SAEs and genuinely non-linear integration patterns that may not be capturable through linear sparse coding approaches.

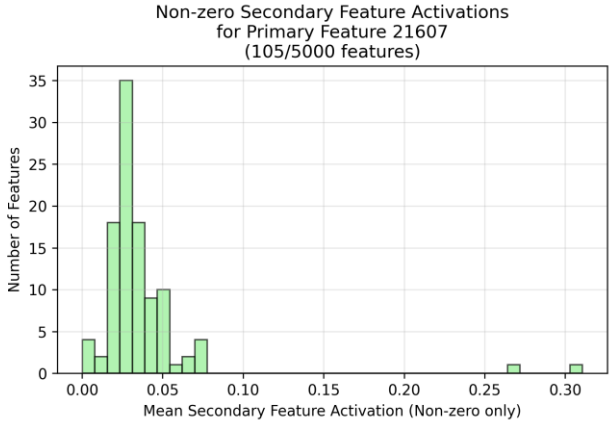
Feature Specificity in Integration Space

Using a stimulus-driven discovery approach, we identified primary SAE features responding to semantic dimensions of our experimental design: Feature 21607 (Emotion) and Feature 21781 (Formality) were selected based on maximal t-test differences across each stimulus conditions (a set of designed input stimuli containing our feature versus a stimuli set lacking that feature, generated by Gemini) directly.

Secondary SAE analysis on the NFM integration pathway revealed selective feature activation patterns. Among 7,500 secondary features, we identified features with distinct sensitivity profiles:

- *Feature 4022*: Highest ANOVA sensitivity (F=26.72, $p=2.85\times10^{-9}$) across experimental conditions
- *Feature 2020*: Highest activation in [formal,emotional] conditions (activation=0.517) but less interaction effects than 4022
- Counterexample features: *Feature 1113* showed no ANOVA sensitivity (F=0.022, p=0.996); *Feature 31* showed no activation differences across conditions

Distribution analysis across secondary features revealed a bimodal pattern: most features showed zero contribution to the primary feature dimensions, while a smaller subset exhibited normal distributions around meaningful contribution levels, with our target features appearing as outliers in the high-contribution tail.



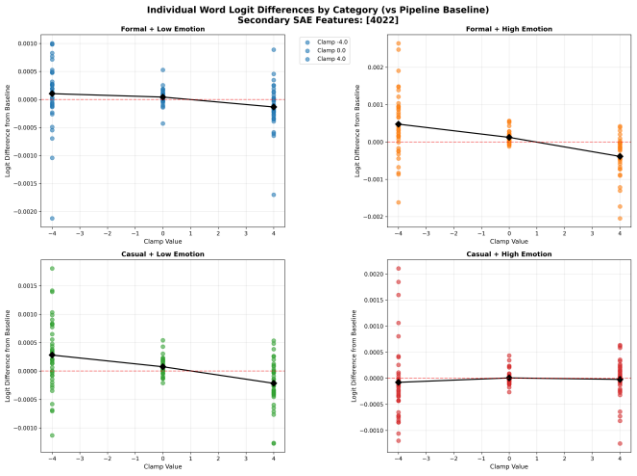
Secondary features (Secondary SAE trained on NFM embedding) mean activation values when only primary feature (21607, Emotion) is naturally activated by our stimulus set. Note: Only non-zero is shown (vast majority are zero), and that the other primary feature (21781, Formality) produces a very similar secondary feature activation distribution.

Intervention Validation and Behavioral Effects

Systematic clamping experiments on *Feature 4022* demonstrated selective behavioral effects across vocabulary categories. Using clamping multipliers of [-4×, 0×, 4×], we measured logit changes for predetermined word sets:

- Formal/low-emotion: "perhaps," "therefore," "consequently"
- Formal/high-emotion: "profoundly," "devastated," "extraordinary"
- Casual/low-emotion: "yeah," "basically," "whatever"
- Casual/high-emotion: "totally," "literally," "absolutely"

Statistical validation confirmed significant interaction effects (F=5.06, p=0.027 for formality×emotion interaction), demonstrating that *Feature 4022* clamping produced non-additive effects across the 2×2 semantic space rather than simple main effects. Additionally, significant effects were seen on the other clamp values (-4.0: p=0.0273, 0.0 p=0.0257, 4.0 p = 0.0397), along with a differential impact of interaction effects by clamping (-4.0: 7e-4, 0.0: 1e-4, 4.0: 4e-4).



X-axis is clamping values, Y-axis are reconstructed logits, four graphs are the 2x2 category of our high/low for our features (Emotion, Formality). Note the differential impact of our secondary interaction feature on bottom right (low Formality, high Emotion) driving the statistically significant interaction in the ANOVA.

Control experiments validated specificity:

- **Linear component clamping**: No interaction effects observed, as expected, from our linear layers, when comparing clamped linear NFM weights vs baseline, across our logit groups.
- **Non-sensitive feature clamping**: Feature 1113 showed no systematic patterns across categories (F>0.28, p>0.18, for all clamping ranges) despite activation.

This counter-factual exploration is not exhaustive but, along with the distribution plots, indicate some specificity of the interaction features discovered by our workflow.

Sequential Architecture KL Divergence Analysis

To further explore whether our improved reconstruction translates to better capturing distributions of logit fidelity compared to typical SAEs, we used Gurnee's test of KL divergence test vs a baseline test of a reconstruction with an ε-random error, reproducing an worse SAE divergence of 3x (compared to Gurnee's reported 2-4.5x range). Our new SAE-NFM sequential architecture showed *29.8% reduction* in pathological KL divergence errors across 17.8M measurements (SAE 2.99x, SAE+NFM 2.1x). Component-specific analysis revealed each of the components were sub-additive but significantly positive (*linear 29.8%, t=806.3; interactive 0.9%, t=1375.8*). These results indicated our approach may address pathological structure in SAE reconstructions.

Joint Training Architecture Performance

Joint training substantially outperformed the sequential approach across reconstruction and behavioral metrics achieving 41.3% reconstruction improvement over the SAE (*joint model reconstruction error: 0.162, SAE reconstruction error: 0.275*), compared to 23% for sequential training.

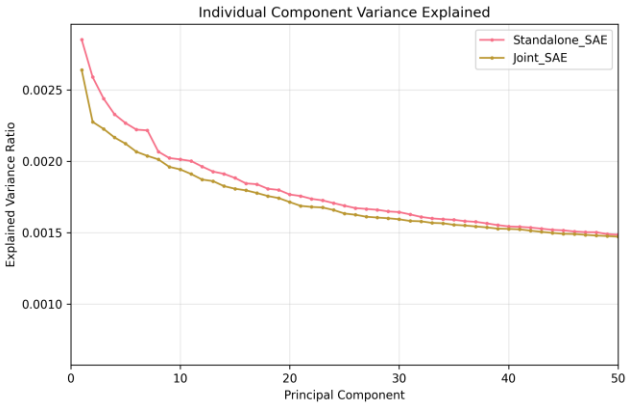
Component analysis showed the NFM linear interaction component comprised 94.0% of NFM parameters by mean absolute embedding weight (*0.347*), while nonlinear interactions comprised 6.0% (*0.022*), by the end of training.

KL Divergence Error Analysis

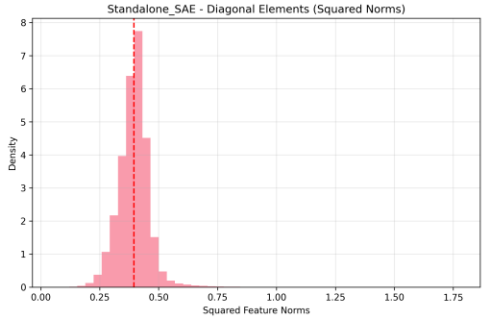
We validated the joint architecture against pathological KL divergence errors using 3.2M measurements. KL divergence by component showed 50.9%, 8.6%, and 51.9% ($t \geq 15.8$, $p < 10e\text{-}6$) for the linear, nonlinear, and combination respectively. Cross-entropy loss showed a similar sub-additive pattern with linear, nonlinear, and combined components achieving 25.7%, 4.2%, and 26.2% improvements respectively ($t \geq 10.3$, $p < 10e\text{-}6$). The nonlinear component alone achieved 16.5% of the total improvement using only 3% of the architecture's total parameters (1.1M, 9% of NFM interaction parameters), demonstrating parameter efficiency in capturing nonlinear computational relationships.

Feature Orthogonality and Emergent Specialization

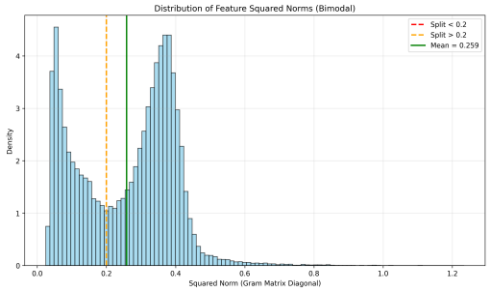
PCA analysis. Principal component analysis showed both architectures achieved 90% variance explained with a similar number of principal components (~860), but the joint architecture required more dimensions across the first 50 components, indicating higher-dimensional feature representations.



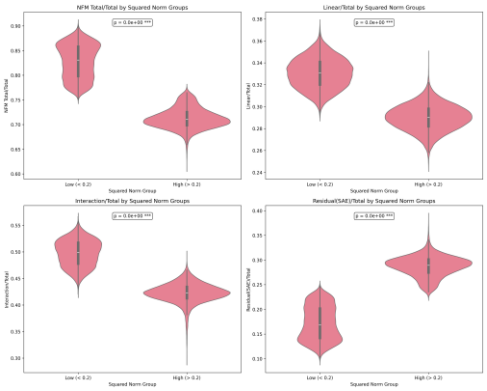
Gram matrix analysis. Analysis of feature weight orthogonality revealed a distinct organizational pattern in the joint architecture versus a standard SAE (*below)* showing a unimodal distribution (mean ~0.4).



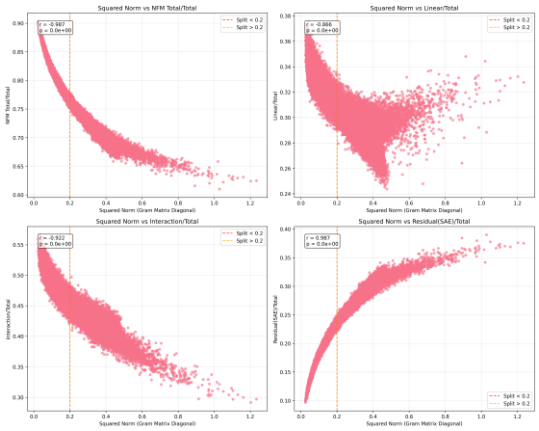
Gram analysis of our joint architecture showed a *bimodal distribution* with clear separation between highly orthogonal features (mean ~0.05) and moderately orthogonal features (mean ~0.37).



Features with higher orthogonality (squared norms <0.2) contributed significantly more to our interaction components overall (*<0.2: 82.8% ,vs >0.2: 71.3%*) by mean absolute weight, and when broken down by the linear interaction component alone (*<0.2: 33.1% ,vs >0.2: 29.0%*), and nonlinear interaction component (*<0.2: 49.7% ,vs >0.2: 42.3%*). While the opposite pattern of direct residual contributions from the SAE were more from our lower orthogonal features (*<0.2: 17.2% ,vs >0.2: 28.7%*).



This creates strong negative correlations (*r=-0.987, -0.867, -0.922, for total, linear, nonlinear*) between our squared norm (feature non-orthogonality) and contributions by weight to our interaction components. Our residual contribution coming only from the SAE encoder has the opposite pattern (*r=0.987*).



These findings suggest the joint architecture effectively forces more orthogonal feature definition in the SAE encoder allowing the linear and nonlinear combinations to occur in our interaction components downstream. It also suggests a secondary distribution of less defined features, potentially because of our limited parameter space, that have to simultaneously service the complex feature interactions along with feature identity, resulting in less orthogonal feature definition. Overall, orthogonality improved compared to standard SAE training.

Discussion

Implications for Neural Computation

Our findings challenge the prevailing view of neural networks as sparse feature storage systems, revealing instead a dual encoding architecture where neural representations simultaneously compress both feature identity and feature integration information. The reconstruction improvement achieved by capturing integration patterns demonstrates that current sparse coding approaches, while successful at identifying interpretable features, systematically miss computational structure that is functionally significant for model behavior.

Polysemantic neurons may not represent compression artifacts to be eliminated, but rather computational units that encode relationships between concepts. The selective intervention effects we observed—where some features produce systematic 2×2 interaction patterns while others show no effects—suggest that polysemanticity may reflect meaningful

computational roles rather than random interference patterns. This reframes the persistent polysemanticity observed even in high-capacity SAEs from a limitation to be overcome to evidence of fundamental computational organization.

The bimodal Gram matrix distribution provides direct empirical evidence for natural computational clustering. Joint training spontaneously develops two distinct feature populations: highly orthogonal features (mean=0.04) specializing in integration pathways, and moderately orthogonal features (mean=0.4) handling direct reconstruction. This emergent organization validates our dual-encoding hypothesis at the representational level—the network naturally separates these computational roles when given architectural flexibility.

Standard SAE training shows unimodal orthogonality distribution (mean≈0.4), suggesting current approaches force all features into a single computational role. The strong positive correlations (r=0.987) between orthogonality and integration contributions fundamentally reframes what orthogonality means in neural representations: rather than indicating independent concepts, orthogonal features may specialize in computational relationships precisely because their clean separation enables effective recombination.

Architectural Integration vs Post-Hoc Analysis

Our comparison of sequential (23% improvement) and joint training (41.3% improvement) methods demonstrates a fundamental architectural principle: integration patterns benefit from simultaneous optimization with identity features rather than post-hoc capture of reconstruction residuals. This represents a paradigm shift from detecting missing computational structure to building architectures that naturally capture it during learning.

Traditional SAE approaches train sparse features first, then attempt to analyze or correct their limitations. Joint training allows the network to develop specialized feature types naturally—orthogonal features that can be effectively recombined in interaction components alongside less orthogonal features that handle complex interactions directly. This architectural flexibility eliminates the need to retrofit integration capture onto pre-trained sparse representations.

The 51.6% KL divergence reduction (vs 30% reduction in our sequential methodology) demonstrates that this architectural approach addresses fundamental limitations rather than providing incremental improvements. Joint training establishes a new standard for SAE architectures that integrate computational relationship modeling from the outset.

Relation to Existing Work

Our framework provides a possibly unifying explanation for several limitations identified in current interpretability research. The "dark matter" described in circuit analysis—computation that remains unexplained despite extensive feature identification (Olah et al. 2020; Sharkey, Braun, and Millidge 2025)—may largely reflect missing integration patterns rather than inadequate feature discovery. Our demonstration that traditional maximum activation approaches fail to identify integration features, despite their clear functional effects, suggests that current interpretability methods may be systematically blind to this form of computation.

Our results provide a direct solution to the pathological KL divergence errors identified by Gurnee et al.[1], achieving 51.6% reduction. This demonstrates that integration capture doesn't merely explain missing computational structure—it systematically addresses the behavioral limitations that have constrained SAE applicability. The pathological nature of these errors reflects missing integration structure rather than random noise, and our architectural approach prevents these systematic failures rather than detecting them post-hoc. The systematic reconstruction errors in logit probability distributions find a natural explanation within our dual encoding framework: these errors reflect missing integration structure rather than random noise or capacity limitations.

Our work also addresses the "wrong abstraction level" problem frequently encountered in SAE research (Chanin, Shlegeris, and Brundage 2024; Makelov et al. 2024; Ayonrinde et al. 2024), where features appear either too specific or too general for interpretable analysis. Under our framework, this may reflect the artificial separation of identity and integration encoding: some apparent features may actually be integration patterns, while some apparent integrations may be underspecified identity features awaiting sufficient encoding capacity.

Unlike static feature relationship methods (Park et al. 2024) that capture co-occurrence patterns, feature integration analysis reveals computational relationships—how features combine to produce emergent meanings that cannot be predicted from their individual activation patterns or statistical co-occurrence. This distinction is crucial for understanding the difference between features that merely appear together and features that compute together.

The parameter efficiency of our integration components (~32.3% of our total architecture's 496M parameters) contributing 40%+ gain in reconstruction loss suggests computational relationships require fundamentally different representational approaches than identity features. Interestingly, our *nonlinear* integration components (3% of parameters achieving 16.5% improvement) provide compelling evidence that nonlinearity in these interactions make substantial contributions to reconstruction performance. Potentially, captured and encoded nonlinear relationships between encoded features, learned by the preceding nonlinear layers of the LLM serve a bigger role than previously thought. Overall, the interaction parameter efficiency indi-

cates that while identity representation may require extensive sparse coding, integration patterns can be captured through targeted architectural components with disproportionate functional impact.

Limitations and Future Work

Scale constraints represent one primary limitation of this work. Our experiments on a 3B parameter model with 50k SAE features provide proof-of-concept evidence, but scaling to industrial-scale models with millions of features remains challenging. The computational requirements of NFM training scale super-linearly with feature count, necessitating architectural innovations or more efficient approximation methods.

Integration interpretability presents ongoing challenges. While we demonstrated functional effects of integration features through systematic interventions, these features remain largely opaque to direct inspection. The failure of maximum activation analysis to yield interpretable patterns for integration features suggests need for specialized interpretability methods designed for computational rather than representational structure.

Limitations of Traditional Feature Discovery

Discovery-oriented feature identification approaches are typically shown in interpretability experiments which has the advantage of being scalable and fairly objective, versus stimulus-oriented feature identification. While several discovery-oriented approaches were attempted, we ran into issues with secondary feature identification. Relatively clean primary features could be identified in our primary SAE, along with corresponding secondary feature indices in which they demonstrated an interaction, but we ran into issues identifying what the secondary feature meant. Additionally, the behavioral output of our small Llama model was not reliable, even when using a primary SAE alone.

Maximum activation analysis on secondary features consistently returned conjunctive tokens ("that") or punctuation features (")") rather than interpretable semantic patterns. This failure occurred despite clear functional effects demonstrated through intervention experiments, suggesting that integration features may not correspond to simple activation maxima in natural text.

These results provide converging evidence that neural networks encode feature integration patterns alongside feature identity, with integration features exhibiting selective sensitivity to experimental manipulations and producing systematic behavioral effects despite their opacity to traditional discovery methods.

These initial findings established proof-of-concept for dual encoding but revealed limitations in the sequential training approach. We therefore developed an integrated joint training architecture to test whether simultaneous optimization could improve both reconstruction fidelity and feature orthogonality. However, there are other possible architectures that may do this better or more efficiently.

The small nonlinear interactions punch above their weight but in their current form (3% of the total architecture parameters), but conflate all higher-order feature combinations (2-way, 3-way, 4-way, etc.) in a compact dense parameter space. Other computationally efficient routes to explore nonlinear interactions without conflation may be advantageous.

Methodological extensions could address several current limitations: (1) Dynamic analysis of how integration patterns evolve during training could reveal the mechanisms by which computational relationships crystallize into identity features. (2) Cross-model validation could establish whether specific integration patterns represent universal computational primitives or model-specific artifacts. (3) Cross-layer analysis could demonstrate the dynamics of feature integration as activity gets processed through layers. (4) Application to larger models could test whether the linear/nonlinear interaction split observed here reflects fundamental properties of neural computation or artifacts of limited scale.

Conclusion

This work provides the first systematic evidence for dual-encoding spaces in neural network representations and introduces an architectural solution that achieves substantial improvements across multiple validation metrics. Joint training delivers 41.3% reconstruction improvement and 51.6% reduction in pathological KL divergence errors while spontaneously developing bimodal feature organization that validates our dual-encoding hypothesis. Critically, systematic intervention experiments revealed integration features with selective sensitivity to experimental manipulations, producing significant interaction effects ($F=5.06$, $p=0.027$) across semantic dimensions rather than simple main effects. The architecture demonstrates that computational relationships can be captured efficiently (32.3% of parameters achieving 41.3% improvement) when separated from identity representation.

This work establishes an architectural paradigm shift from post-hoc feature analysis to integrated computational design. Rather than training sparse autoencoders and then analyzing their limitations, joint training enables natural specialization where orthogonal features form cleaner definitions that can be more effectively utilized by specialized interaction components, while less orthogonal features must serve multiple computational purposes with less precise definitions. This emergent organization—evidenced through bimodal Gram matrix distributions and systematic specialization patterns—demonstrates that networks naturally separate identity and integration encoding when given appropriate architectural flexibility.

Methodological contributions include the first demonstration of separable feature identity and integration encoding, systematic approaches for detecting and intervention on the computational components that combine features, and stimulus-oriented validation methodologies that may have advantages for establishing functional significance beyond discovery-oriented approaches.

Broader implications extend beyond interpretability to fundamental questions about neural computation, AI safety, and the relationship between artificial and biological neural systems. Understanding how networks integrate information to produce emergent behaviors is crucial for developing reliable, controllable AI systems and for advancing theories of intelligence itself.

The convergence of reconstruction improvements, behavioral validation, and mechanistic understanding positions this approach as a foundation for next-generation sparse autoencoder architectures. By solving known limitations (pathological errors), providing architectural innovation (joint training), and revealing natural computational organization (emergent specialization), this work advances both the theoretical understanding and practical implementation of neural network interpretability.

References

Ayonrinde, K.; Shah, R.; Fry, S.; Winsor, E.; Gurnee, W.; Tegmark, M.; and Krueger, D. 2024. Interpretability as Compression: Reconsidering SAE Explanations of Neural Activations with MDL-SAEs. arXiv preprint arXiv:2410.11179.

Bereska, L. F.; and Gavves, E. 2024. Mechanistic Interpretability for AI Safety — A Review. arXiv preprint arXiv:2407.11215.

Bricken, T.; Templeton, A.; Batson, J.; Chen, B.; Jermyn, A.; Conerly, T.; Turner, N. L.; Anil, C.; Denison, C.; Askell, A.; Lasenby, R.; Wu, Y.; Kravec, S.; Schiefer, N.; Maxwell, T.; Joseph, N.; Tamkin, A.; Nguyen, K.; McLean, B.; Burke, J. E.; Hume, T.; Carter, S.; Henighan, T.; and Olah, C. 2023. Towards Monosemanticity: Decomposing Language Models With Dictionary Learning. Transformer Circuits Thread.

Burns, C.; Ye, H.; Klein, D.; and Steinhardt, J. 2022. Discovering Latent Knowledge in Language Models Without Supervision. arXiv preprint arXiv:2212.03827.

Bussmann, B.; Nabeshima, N.; Karvonen, A.; and Nanda, N. 2025. Learning Multi-Level Features with Matryoshka Sparse Autoencoders. arXiv preprint arXiv:2503.17547.

Chanin, D.; Shlegeris, B.; and Brundage, M. 2024. A Is for Absorption: Studying Feature Splitting and Absorption in Sparse Autoencoders. arXiv preprint arXiv:2409.14507.

Chen, S.; Trojanowski, S.; Karpinska, M.; Pavlick, E.; and Bowman, S. R. 2024. Taming Polysemanticity in LLMs: Provable Feature Recovery via Sparse Autoencoders. arXiv preprint arXiv:2506.14002.

Cunningham, H.; Ewart, T.; Riggs, L.; Huben, R.; and Sharkey, L. 2023. Sparse Autoencoders Find Highly Interpretable Features in Language Models. arXiv preprint arXiv:2309.08600.

Elhage, N.; Hume, T.; Olsson, C.; Schiefer, N.; Henighan, T.; Kravec, S.; Hatfield-Dodds, Z.; Lasenby, R.; Drain, D.; Chen, C.; Grosse, R.; McCandlish, S.; Kaplan, J.; Amodei, D.; Wortsman, M.; and Ludwig, J. 2022. Toy Models of Superposition. Transformer Circuits Thread.

Engels, J.; Liao, I.; Michaud, E. J.; Gurnee, W.; and Tegmark, M. 2024. Decomposing The Dark Matter of Sparse Autoencoders. arXiv preprint arXiv:2410.14670.

Field, D. J. 1994. What Is the Goal of Sensory Coding? Neural Computation 6(4): 559–601. doi.org/10.1162/neco.1994.6.4.559.

Gao, L.; Schulman, J.; and Hilton, J. 2024. Scaling and Evaluating Sparse Autoencoders. arXiv preprint arXiv:2406.04093.

Makelov, A.; Sharma, M.; Tong, M.; Hernandez, E.; Braun, J.; Pehlevan, C.; and Tegmark, M. 2024. Towards Principled Evaluations of Sparse Autoencoders for Interpretability and Control. arXiv preprint arXiv:2405.08366.

Nanda, N.; Chan, L.; Lieberum, T.; Smith, J.; and Steinhardt, J. 2023. Progress Measures for Grokking via Mechanistic Interpretability. arXiv preprint arXiv:2301.05217.

Olah, C.; Cammarata, N.; Schubert, L.; Goh, G.; Petrov, M.; and Carter, S. 2020. Zoom In: An Introduction to Circuits. Distill 5(3): e00024.001. doi.org/10.23915/distill.00024.001.

Olah, C.; Mordvintsev, A.; and Schubert, L. 2017. Feature Visualization. Distill 2(11): e7. doi.org/10.23915/distill.00007.

Olshausen, B. A.; and Field, D. J. 1996. Emergence of Simple-Cell Receptive Field Properties by Learning a Sparse Code for Natural Images. Nature 381(6583): 607–609. doi.org/10.1038/381607a0.

Park, K.; Ro, Y.; Liu, H.; and Kim, J. 2024. The Linear Representation Hypothesis and the Geometry of Large Language Models. arXiv preprint arXiv:2311.03658.

Sharkey, L.; Braun, D.; and Millidge, B. 2025. Open Problems in Mechanistic Interpretability. arXiv preprint arXiv:2501.16496.

Tamkin, A.; Jurafsky, D.; and Goodman, N. 2023. Codebook Features: Sparse and Discrete Interpretability for Neural Networks. arXiv preprint arXiv:2310.17230.

Templeton, A.; Conerly, T.; Marcus, J.; Lindsey, J.; Bricken, T.; Chen, B.; Pearce, A.; Citro, C.; Ameisen, E.; Jones, A.; Cunningham, H.; Turner, N. L.; McDougall, C.; MacDiarmid, M.; Tamkin, A.; Durmus, E.; Hume, T.; Mosconi, F.; Freeman, C. D.; Sumers, T. R.; Rees, E.; Batson, J.; Jermyn, A.; Carter, S.; Olah, C.; and Henighan, T. 2024. Scaling Monosemanticity: Extracting Interpretable Features from Claude 3 Sonnet. Transformer Circuits Thread.

Footnotes

[1] Gurnee, W. "SAE reconstruction errors are (empirically) pathological." AI Alignment Forum, March 29, 2024. https://www.alignmentforum.org/posts/rZPiuFxESMxCDHe4B/sae-reconstruction-errors-are-empirically-pathological

[2] Olah, C. "Interpretability Dreams." Transformer Circuits Thread, May 24, 2023. https://transformer-circuits.pub/2023/interpretability-dreams/index.html