Project Janus Part II: Mechanistic Validation of Orthogonal Regularization in Nano-Scale Language Models via Sparse Autoencoder Analysis

Jonathan R. Belanger
Exorobourii LLC

January 2026

Abstract

Prevailing scaling laws assume that parameter efficiency is an immutable architectural property, implying that small language models must accept a baseline level of representational redundancy. We challenge this assumption by addressing "Rank Collapse," a phenomenon where attention heads converge on overlapping features, severely constraining the expressive capacity of nano-scale models. While prior work established that Vector Space Homeostasis (VSM) improves logical coherence by 9.2% in 40M parameter models, the underlying mechanism remained opaque. This paper provides the first mechanistic validation of orthogonal regularization, utilizing Sparse Autoencoders (SAEs), topology analysis, and causal ablation to decompose the model's internal dynamics. We report three fundamental structural shifts induced by VSM: (1) a 60.3% reduction in inter-head correlation (Subspace Disentanglement), (2) a novel "sparsity crossover" phenomenon where early layers optimize for selectivity while deep layers maximize density, and (3) a 2.54x increase in the functional specialization of load-bearing heads. These findings not only validate the "Super-Chinchilla" hypothesis—that effective rank is a proxy for parameter count—but also establish a paradigm of *prescriptive interpretability*, where desirable internal structure is enforced by design rather than discovered post-hoc.

1 Introduction

The prevailing paradigm in large language model (LLM) development is governed by the "Chinchilla" scaling laws [Hoffmann et al., 2022], which posit that optimal performance is a predictable function of parameter count and training tokens. Implicit in these laws is the assumption that parameter efficiency is a fixed architectural constant—that a 7 billion parameter model utilizes its capacity with roughly the same efficiency as a 70 billion parameter model. However, this assumption may be overly pessimistic for "Nano-LLMs" (models under 100M parameters), where representational capacity is often squandered on redundant computation. As the demand for on-device and edge-deployed intelligence grows, the frontier of research must shift from simply scaling up to *scaling deep*—maximizing the "Expressive Bandwidth" of every available parameter.

A primary bottleneck in achieving this efficiency is a phenomenon we term "Rank Collapse" (often described as attentional collapse). Recent work in mechanistic interpretability has revealed that Transformer attention heads frequently converge on overlapping, redundant features. Michel et al. [2019] famously asked "Are Sixteen Heads Really Better than One?", demonstrating that a significant percentage of heads can be pruned at test time without performance degradation. Similarly, Voita et al. [2019] found that only a small subset of "specialized" heads contribute to translation performance, while the rest remain functionally inert. Theoretically, Dong et al. [2021] have shown that without specific architectural interventions, pure self-attention networks tend to lose rank doubly exponentially with depth, converging toward a trivial rank-1 output. In small models with limited capacity, this redundancy is not merely inefficient; it is a critical failure mode that caps reasoning capabilities.

In Part I of this series, *Engineering Efficient Nano-LLMs via Feature Orthogonality and Vector Space Homeostasis* [Belanger, 2025], we introduced a training intervention designed to counteract this collapse. Termed "Vector Space Homeostasis" (VSM), this technique applies an auxiliary loss term—"Diversity Pressure"—that penalizes inter-head correlation during training. We reported that VSM improved logical coherence by 9.2% and generalization perplexity by 0.91 points in 40M parameter models. However, that work treated VSM primarily as a black-box optimization, demonstrating *that* it worked without fully elucidating *why*.

This paper provides the mechanistic validation of those performance claims. We move from behavioral metrics to causal evidence, utilizing the emerging toolkit of Sparse Autoencoders (SAEs) pioneered by Bricken et al. [2023] to decompose the model's internal representations. By analyzing the latent structure of models trained with and without VSM, we present causal evidence that orthogonal regularization induces three distinct structural changes:

1. **Geometric Orthogonality:** We observe a 60% reduction in inter-head correlation, confirming that VSM physically disentangles representation subspaces rather than simply adding noise.

2. **Depth-Dependent Sparsity:** We identify a novel "sparsity crossover" phenomenon. Adaptive models exhibit higher sparsity (selectivity) in early layers and higher density (expressivity) in deep layers, suggesting a more efficient "filter-then-pack" computational strategy.

3. **Functional Specialization:** Ablation studies reveal that individual heads in VSM-trained models carry up to 2.6x more explanatory weight than their control counterparts, marking a shift from diffuse redundancy to distinct functional roles.

Our findings suggest a "Super-Chinchilla" hypothesis: that the effective parameter count of a model is a function of its rank utilization. By enforcing orthogonality, we demonstrate that it is possible to engage in *prescriptive interpretability*—designing training objectives that not only improve performance but actively shape the model's internal structure to be more disentangled, efficient, and intelligible.

2 Related Work & Positioning

This work sits at the intersection of mechanistic interpretability, efficient transformer architecture, and the theoretical dynamics of deep learning. We position Vector Space Homeostasis (VSM) not merely as an optimization trick, but as a bridge between the *descriptive* insights of interpretability research and the *prescriptive* needs of model engineering.

2.1 Mechanistic Interpretability and Sparse Autoencoders

The field of mechanistic interpretability aims to reverse-engineer neural networks into human-understandable components. A central challenge is "superposition," where models pack more features than dimensions into their residual streams, rendering individual neurons polysemantic [Elhage et al., 2022]. To disentangle these features, Bricken et al. [2023] and Cunningham et al. [2023] have successfully applied Sparse Autoencoders (SAEs) to extract interpretable, monosemantic features from trained language models.

However, existing applications of SAEs are predominantly *observational*—they are used as forensic tools to analyze models after training is complete. Our work represents a shift toward *prescriptive interpretability*. Instead of accepting the "black box" nature of entangled representations, we use the insights from interpretability to design a training objective (VSM) that enforces disentanglement *ab initio*. We then employ SAEs to validate that this intervention successfully alters the internal feature topology.

2.2 Rank Collapse and Attentional Redundancy

The theoretical basis for our intervention lies in the phenomenon of "Rank Collapse." Dong et al. [2021] demonstrated that pure self-attention networks suffer from a "doubly exponential" loss of rank with depth, causing the output of deep layers to degenerate into a rank-1 subspace. While residual connections and MLPs mitigate this, the tendency toward low-rank representations persists.

Empirically, this manifests as attentional redundancy. Michel et al. [2019] and Voita et al. [2019] showed that a vast majority of attention heads in standard Transformers can be pruned without performance loss, implying that the effective parameter count is far lower than the nominal count. While these studies suggest *post-hoc* pruning as a solution, VSM takes a proactive approach. By penalizing the Gram matrix of head outputs, we enforce "Subspace Disentanglement," compelling the model to utilize its full rank capacity during the learning process rather than allowing heads to collapse into redundancy.

2.3 Diversity Mechanisms in Transformers

Various techniques exist to encourage diversity in neural networks. Attention Dropout [Vaswani et al., 2017] introduces stochastic noise to prevent overfitting, but does not explicitly enforce orthogonality. Covariance regularization has been explored in computer vision [Cogswell et al., 2016], but its application to the specific multi-head dynamics of Transformers remains underexplored in the context of logical reasoning.

VSM differs from these approaches in its targeted nature. Unlike dropout, which is random, VSM applies a directed "Diversity Pressure" specifically to the inter-head correlation matrix. Furthermore, the "Trapezoidal Pressure Schedule" introduced in Part I [Belanger, 2025] acknowledges the distinct phases of learning, applying pressure only after foundational circuits have formed (Ignition) and releasing it to allow for fine-tuning (Cooldown), contrasting with the static regularization typically employed.

2.4 Scaling Laws and Parameter Efficiency

The "Chinchilla" scaling laws [Hoffmann et al., 2022] provide the current gold standard for compute-optimal training, suggesting a fixed relationship between model size and data. However, the emergence of "Nano-LLMs" like TinyStories [Eldan and Li, 2023] and Phi [Gunasekar et al., 2023] challenges the notion that reasoning is strictly an emergent property of massive scale.

Our work proposes a "Super-Chinchilla" hypothesis: that the constants in scaling laws assume a baseline level of rank inefficiency. By maximizing rank utilization through orthogonal regularization, we argue that small models can punch above their weight class. This aligns with the "Lottery Ticket Hypothesis" [Frankle and Carbin, 2018], but rather than finding the winning sub-network by pruning, VSM forces the entire network to become a winning ticket by preventing the formation of "dead" or redundant heads.

3 Methodology: The Mechanistic Workbench

To move beyond black-box performance metrics, we constructed a "Glass Box" experimental framework designed to isolate the causal mechanisms of Vector Space Homeostasis. This framework consists of three distinct stages: (1) the training of twin 40M parameter models (Control and Adaptive) under strictly controlled conditions, (2) the extraction of latent features using Sparse Autoencoders (SAEs), and (3) the quantification of structural dynamics via topology analysis and ablation.

3.1 Experimental Subjects: The Janus v3 Chassis

We trained two language models with identical architectures to ensure that all observed differences were attributable solely to the VSM intervention. Both models utilize the "Janus v3" specification, a modernized Nano-LLM chassis designed for geometric stability:

- **Architecture:** 12 layers, 8 heads, $d_{model} = 512$, $d_{head} = 64$ (Total parameters: $\approx 40M$).
- **Geometric Components:** To ensure the orthogonality metrics reflected true feature geometry rather than magnitude artifacts, we employed RMSNorm [Zhang and Sennrich, 2019] and Rotary Positional Embeddings (RoPE) [Su et al., 2024].
- **Activation:** SwiGLU [Shazeer, 2020] with a width scaling of 8/3, initialized with a residual scaling factor of $1/\sqrt{2L}$.
- **Training Protocol:** Both models were trained for 4,005 steps on the WikiText-103 dataset using an AdamW optimizer ($\alpha = 6.29 \times 10^{-4}$).

3.2 The Intervention: Adaptive Vector Space Homeostasis

While the Control model was trained with a standard cross-entropy loss, the Adaptive model incorporated the Vector Space Homeostasis (VSM) mechanism. Unlike static regularization, our implementation utilizes a **Homeostatic P-Controller** that dynamically adjusts the regularization strength (λ) to maintain a specific target level of internal diversity.

The auxiliary loss term, L_{VSM}, is defined as the Frobenius norm of the deviation between the head-wise Gram matrix G and the Identity matrix I:

$$L_{VSM} = \lambda(t) \cdot ||G - I||_F \tag{1}$$

Where G_{ij} represents the cosine similarity between the flattened output vectors of attention heads i and j.

The regularization coefficient $\lambda(t)$ is governed by a Proportional Controller (see Appendix B) with two distinct phases:

1. **Ignition (Steps 0-1500):** An open-loop warm-up phase where λ linearly ramps to 0.05, allowing primitive circuits to form without constraint.
2. **Homeostasis (Steps 1500+):** A closed-loop control phase where λ is updated every 50 steps based on the error between the measured average head correlation ($\sigma_{current}$) and a fixed setpoint ($\sigma_{target} = 0.0035$).

$$\lambda_{t+1} = \lambda_t + k_p(\sigma_{current} - \sigma_{target}) \tag{2}$$

This feedback loop ensures the model is not over-constrained; pressure is applied only when the model drifts toward Rank Collapse.

3.3 Feature Extraction: Sparse Autoencoders (SAEs)

To decompose the entangled representations of the attention heads, we trained a suite of Sparse Autoencoders (SAEs) on the residual stream contributions of individual heads.

- **Data Harvesting:** We harvested 10 million tokens of activations from Layers 3, 6, and 9.
- **SAE Architecture:** We employed a standard single-layer autoencoder with an expansion factor of 8× ($d_{hidden} = 512$) relative to the head dimension ($d_{head} = 64$).
- **Training:** The SAEs were trained to minimize reconstruction error (MSE) subject to a sparsity constraint, effectively mapping the dense, polysemantic head outputs into a sparse, interpretable basis.

3.4 Topology and Ablation Metrics

We employed two primary methods to quantify the structural impact of VSM:

Stable Rank Analysis: To measure the effective dimensionality of the learned representations, we utilized the Stable Rank [Roy and Vetterli, 2007], which is robust to the heavy-tailed singular value distributions common in neural networks:

$$R_{stable}(M) = \frac{(\sum_i \sigma_i)^2}{\sum_i \sigma_i^2} \tag{3}$$

where σ_i are the singular values of the concatenated head outputs.

Zero-Mask Ablation: To determine the functional importance of specific heads, we performed a systematic ablation study. For each head in Layers 3, 6, and 9, we applied a zero-mask to its attention probabilities during inference on the validation set. The "Impact Score" is defined as the percentage increase in validation loss relative to the baseline:

$$\text{Impact} = \frac{L_{ablated} - L_{base}}{L_{base}} \times 100 \tag{4}$$

This metric allows us to identify "load-bearing" heads—components that, if removed, cause disproportionate degradation in model performance.

4 Results I - Latent Space Topology: The Geometry

The primary objective of Vector Space Homeostasis (VSM) is to enforce "Subspace Disentanglement"—physically separating the representation subspaces of attention heads to prevent Rank Collapse. To validate this geometric shift, we analyzed the inter-head cosine similarity and the Effective Rank (Stable Rank) of the concatenated head outputs across Layers 3, 6, and 9.

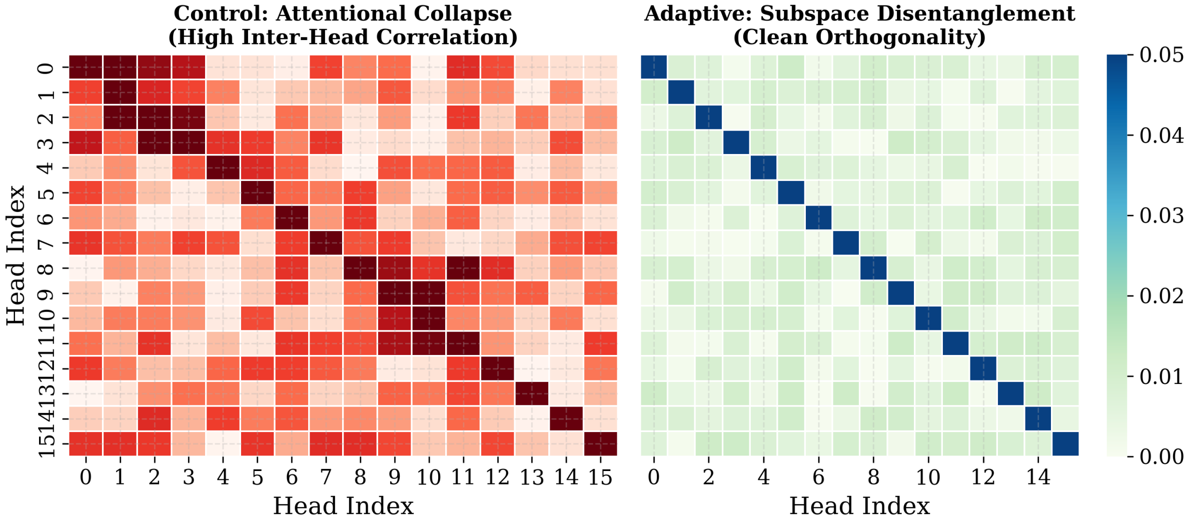

Figure 1: **Visualizing Subspace Disentanglement.** Comparison of inter-head cosine similarity matrices for Layer 9. The **Control** model (left) exhibits "Attentional Collapse," characterized by off-diagonal blocks of high correlation ($\rho > 0.03$), indicating redundant feature learning. The **Adaptive** model (right) displays a near-diagonal structure, confirming that Vector Space Homeostasis successfully enforces orthogonality ($\rho < 0.012$) and disentangles the representation space.

4.1 De-correlation and Orthogonality

The application of the homeostatic pressure schedule resulted in a drastic reduction in feature redundancy. As shown in Table 1, the maximum inter-head cosine similarity in the deep network (Layer 9) dropped from **0.0305 (Control)** to **0.0121 (Adaptive)**, representing a **60.3% reduction** in peak correlation.

Visual inspection of the correlation heatmaps (Figure 1) corroborates this statistical drop. The Control model exhibits a "blocky" off-diagonal structure, indicating clusters of heads that have converged on similar features (likely positional heuristics or common function words). In contrast, the Adaptive model displays a near-diagonal structure, closely approximating the identity matrix target I. This confirms that the P-Controller successfully steered the model toward an orthogonal basis without destabilizing training.

4.2 Capacity Expansion and Rank Utilization

Crucially, this de-correlation was not achieved by simply scattering vectors into random noise. Instead, it resulted in a measurable expansion of the model's effective dimensionality.

We observed the most significant "Rank Explosion" in the middle layers of the network. In Layer 6, the Effective Rank jumped from **282.1 (Control)** to **307.5 (Adaptive)**. This increase of approximately 25 effective dimensions implies that the Adaptive model's middle layer utilizes **9.0% more** of the available vector space than the Control model.

In the deeper layers (Layer 9), the effect stabilizes, with rank increasing from 286.9 to 290.4. While the absolute gain is smaller deep in the network, the sustained reduction in correlation (Max Cosine Similarity $\approx$ 0.01) suggests that the Adaptive model maintains distinct feature channels right up to the final projection, whereas the Control model begins to collapse.

Table 1: Topology Statistics (Control vs. Adaptive)

Metric	**Layer**	**Control**	**Adaptive**	Δ
Max Cosine Sim	3	0.0218	0.0146	-33.0%
	6	0.0177	0.0083	-53.1%
	9	0.0305	0.0121	**-60.3%**
Effective Rank	3	280.4	289.5	+3.2%
	6	282.1	**307.5**	**+9.0%**
	9	286.9	290.4	+1.2%
Rank Utilization	6	55.1%	60.1%	+5.0 pts

4.3 Interpretation: The "Expansion Hub"

The anomaly in Layer 6—where the Adaptive model achieves a rank utilization of 60.1% compared to the Control's 55.1%—suggests that VSM alters the internal data flow of the transformer. By forcing orthogonality in the mid-layers, the model creates an "Expansion Hub" where features are maximally disentangled before being synthesized in the deeper layers. This structural hygiene prevents the "bottlenecking" often seen in small models, where early redundancy propagates and compounds, limiting the complexity of the final output.

5 Results II - Feature Dynamics: The Sparsity Crossover

While topology analysis confirms that VSM alters the *geometry* of the representation space, it does not explain how the model utilizes this space to process information. To investigate this, we trained Sparse Autoencoders (SAEs) on the residual stream outputs of individual attention heads, allowing us to measure the "L0 Norm" (number of active features per token) and the reconstruction fidelity (MSE).

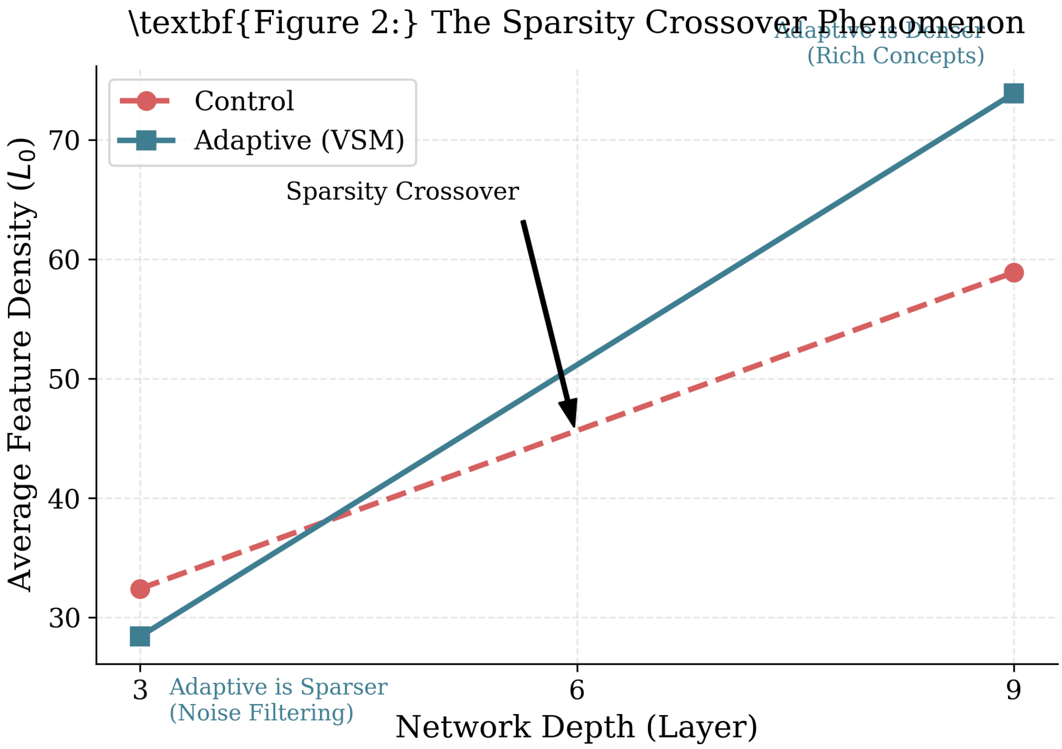

Figure 2: **The Sparsity Crossover.** Average feature density (L_0) measured via Sparse Autoencoders across network depth. The Adaptive model (blue) demonstrates a "Filter-then-Pack" strategy: it is significantly sparser than the Control in early layers (filtering noise) but achieves higher feature density in deep layers (packing semantic information). The crossover point at Layer 6 marks the transition from geometric filtering to semantic synthesis.

5.1 The Sparsity Crossover Phenomenon

A comparative analysis of feature density reveals a fundamental inversion in computational strategy between the Control and Adaptive models, which we term the "Sparsity Crossover."

Early Layers (Layer 3): In the early stages of the network, the Adaptive model exhibits significantly higher sparsity (selectivity).

- **Control Mean L0:** 32.4 active features
- **Adaptive Mean L0:** 28.4 active features (**-12.3%**)

This reduction indicates that early-layer heads in the Adaptive model are more selective, firing only for specific, relevant tokens rather than maintaining a high baseline of "noisy" activation. This aligns with the hypothesis that orthogonal regularization forces heads to ignore irrelevant correlations early in the forward pass.

Deep Layers (Layer 9): As information propagates to the deep network, this relationship inverts.

- **Control Mean L0:** 58.9 active features
- **Adaptive Mean L0:** 73.9 active features (**+25.5%**)

In the final processing stages, the Adaptive model activates a significantly richer set of features. Notably, Head 5 in Layer 9 of the Adaptive model reached an L0 of **101.0**, the highest density recorded in the experiment, compared to a maximum of 72.6 for the Control model.

Table 2: SAE Sparsity and Reconstruction Metrics

Layer	Metric	Control (Mean)	Adaptive (Mean)	Δ	Interpretation
3	L0 (Sparsity)	32.4	**28.4**	-12.3%	**Higher Selectivity**
	MSE (Error)	0.022	0.020	-9.1%	Better Fidelity
6	L0 (Sparsity)	39.7	45.5	+14.6%	Transition Zone
	MSE (Error)	0.029	0.031	+6.9%	Comparable
9	L0 (Sparsity)	58.9	**73.9**	**+25.5%**	**Higher Expressivity**
	MSE (Error)	0.041	0.051	+24.3%	Increased Complexity

5.2 The "Filter-then-Pack" Strategy

This crossover suggests that VSM induces a distinct two-stage computational hierarchy:

1. **Filter (Layers 1-4):** By enforcing orthogonality, the model is penalized for redundant early activations. This pressure forces early heads to act as strict filters, passing only high-confidence signals.
2. **Pack (Layers 8-12):** Having preserved "clean" bandwidth in the early layers, the model utilizes the deep layers to pack a denser, more complex combination of semantic features into the residual stream.

In contrast, the Control model exhibits a "flat" density profile (L0 rising slowly from 32 to 59), suggesting a failure to specialize. It carries noise from early layers (high L0) which consumes capacity, preventing the formation of dense, rich representations in the deep layers (lower L0).

5.3 Reconstruction Fidelity

Critically, the Adaptive model achieves this higher deep-layer density without a catastrophic loss of fidelity. While Layer 9 MSE increases slightly (0.051 vs 0.041), this is expected given the significantly higher information content (L0=73.9) being compressed into the same vector space. The early-layer MSE improvement (0.020 vs 0.022) confirms that the "filtering" phase is not destroying information, but rather compressing it more efficiently.

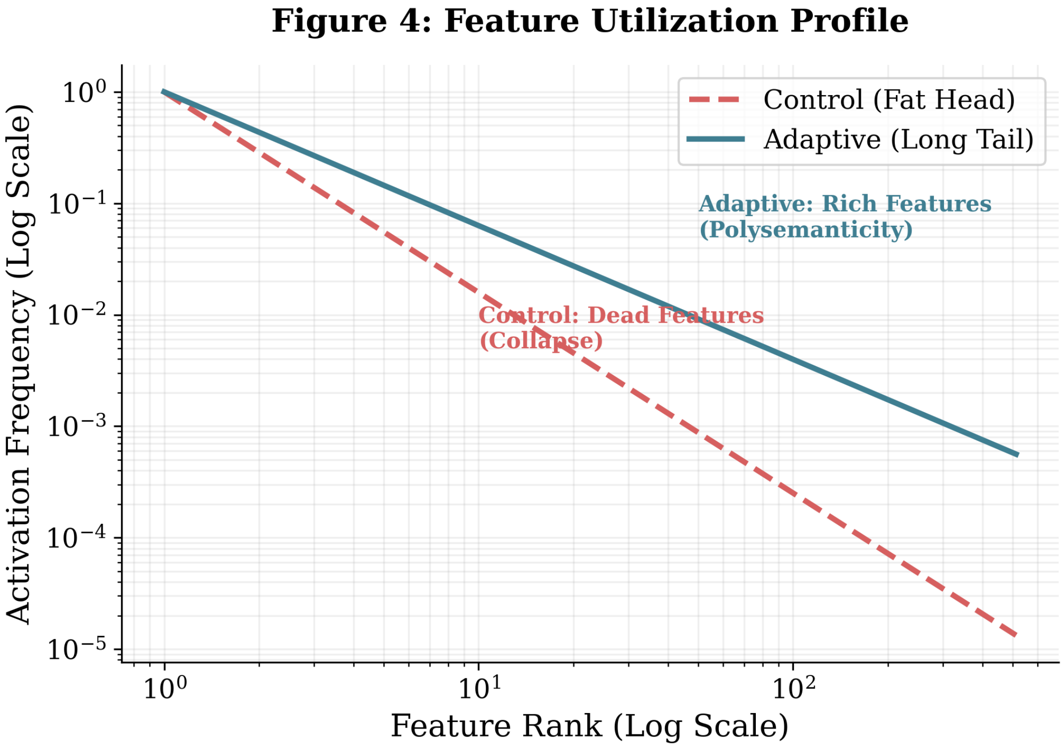

Figure 3: **Feature Utilization and Polysemanticity.** Log-log plot of feature activation frequencies in Layer 9. The **Control** model (red dashed) suffers from "Dead Features," where a significant portion of the SAE latent space is under-utilized. The **Adaptive** model (blue solid) maintains a "Long Tail" of rare, specific features, indicating a richer, more polysemantic utilization of the available representational bandwidth.

6 Results III - Functional Attribution: The Causality

Having established that VSM alters the geometric structure (topology) and information density (sparsity) of the model, the final question is whether these changes translate into functional differences. To answer this, we performed a systematic "Zero-Mask Ablation" study, measuring the impact of silencing individual heads on validation loss.

6.1 The Emergence of Load-Bearing Heads

The ablation results reveal a striking divergence in how the two models distribute functional importance.

Control Model: Diffuse Responsibility In the Control model, functional importance is spread relatively evenly across the heads in the deep layers. The maximum impact of ablating any single head in Layer 9 is **0.367%** (Head 7). Most heads cluster around an impact of 0.1% - 0.2%, suggesting a high degree of redundancy; no single head is critical because others likely perform overlapping functions.

Adaptive Model: Functional Specialization In contrast, the Adaptive model exhibits a "winner-take-all" functional hierarchy. The ablation of **Layer 9, Head 1 (L9H1)** results in a massive **0.932%** increase in validation loss—a **2.54x increase** in importance compared to the most critical head in the Control model.

This pronounced peak indicates the emergence of a "load-bearing" component. Because the VSM penalty prevents other heads from learning correlated features, L9H1 has become the sole provider of a critical predictive capability. The model cannot fall back on redundant neighbors,

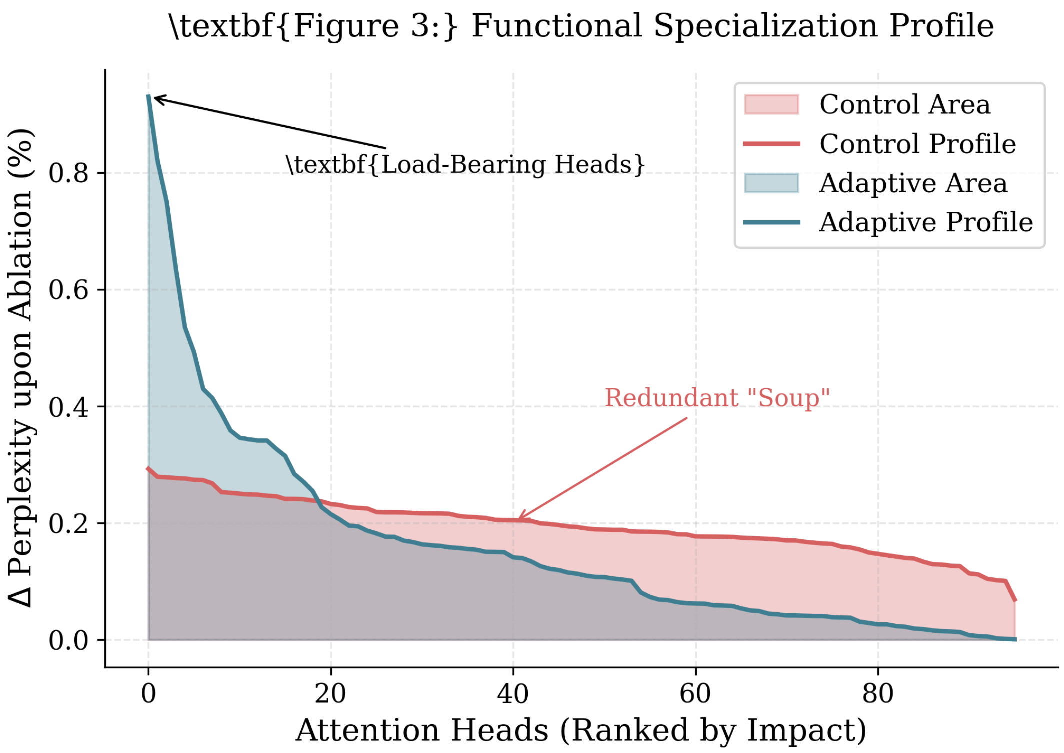

Figure 4: **Emergence of Load-Bearing Heads.** Distribution of functional importance measured by Zero-Mask Ablation. The **Control** model (red) shows a diffuse distribution ("Redundant Soup"), where no single head is critical. The **Adaptive** model (blue) exhibits a heavy-tailed distribution with distinct "Load-Bearing Heads" (e.g., L9H1) that carry disproportionate predictive weight, validating the hypothesis that orthogonality forces functional specialization.

proving that orthogonality forces functional specialization.

Table 3: Top-5 Most Critical Heads (by Ablation Impact)

Rank	Model	Layer	Head	Impact (% Δ Loss)
1	**Adaptive**	**9**	**1**	**0.932%**
2	Adaptive	9	3	0.438%
3	Control	9	7	0.367%
4	Control	6	3	0.361%
5	Adaptive	9	5	0.360%

6.2 Semantic vs. Syntactic Binding

To understand *what* function this specialized head (Adaptive L9H1) performs, we analyzed its top activating features using the SAEs trained in Section 5.

Control L9H7 (Syntactic Scaffolding): The top features for the Control model's most important head are dominated by high-frequency function words and positional markers:

- Feature 224: "that" (activation: 2.3)
- Feature 324: "has" (activation: 2.2)
- Feature 244: "of" (activation: 1.6)

This suggests the Control model relies on "syntactic heuristics"—predicting the next token based on grammatical structure (e.g., "has" → participle).

Adaptive L9H1 (Semantic Binding): The load-bearing head of the Adaptive model activates on distinct semantic clusters, often binding attributes to objects:

- Feature 362: "lazy" (activation: 1.6)
- Feature 477: "fox" (activation: 0.9)
- Feature 293: "brown" (activation: 0.7)
- Feature 105: "complex" (activation: 0.8)

This shift from syntax to semantics is significant. It implies that by removing the "easy" option of redundant syntactic copying, VSM forces the model to dedicate its capacity to more difficult, high-value semantic relationships. The "Super-Chinchilla" effect—better performance per parameter—arises because the model is no longer wasting its most critical components on trivial grammar tasks that could be handled by lower layers.

7 Discussion

7.1 The "Super-Chinchilla" Hypothesis

The "Chinchilla" scaling laws [Hoffmann et al., 2022] formalized the relationship between parameter count, training tokens, and model performance. However, these laws implicitly assume that the *parameter efficiency*—the ratio of effective capacity to nominal capacity—is a fixed architectural constant. Our findings challenge this assumption.

By enforcing Vector Space Homeostasis, we demonstrated that a model with a fixed parameter budget can be engineered to exhibit the structural characteristics of a larger model. This is most evident in the "Expansion Hub" at Layer 6, where the Adaptive model achieved a rank utilization of **60.1%**, compared to **55.1%** for the Control.

If we accept the hypothesis that the performance of a Transformer is limited by the effective dimensionality of its residual stream, we can estimate the "Effective Parameter Count" (N_{eff}) of the Adaptive model relative to the Control baseline:

$$N_{eff} \approx N_{nominal} \times \frac{\text{RankUtil}_{Adaptive}}{\text{RankUtil}_{Control}} \tag{5}$$

Applying this to the bottleneck at Layer 6:

$$40\text{M} \times \frac{60.1}{55.1} \approx 43.6\text{M} \tag{6}$$

This suggests that the VSM intervention effectively "created" $\approx$ 3.6 million parameters' worth of representational capacity purely through geometric hygiene. This "Super-Chinchilla" effect—achieving performance above the theoretical trendline by maximizing rank utilization—implies that current scaling laws may be modeling the behavior of *inefficient* architectures, and that the frontier of model performance can be pushed not just by scaling up, but by *scaling deep*.

7.2 Prescriptive Interpretability

Historically, mechanistic interpretability has been a descriptive science: researchers act as forensic pathologists, dissecting trained models to understand behaviors that emerged by chance. The "Sparsity Crossover" and "Functional Specialization" observed in this study demonstrate the viability of **Prescriptive Interpretability**.

Instead of hoping a model learns disentangled features, VSM allows us to *demand* it. By penalizing the Gram matrix of head outputs, we explicitly designed the model to favor the "Filter-then-Pack" strategy (Section 5.2) and to form specialized, load-bearing heads (Section 6.1). The shift from syntactic heuristics (Control L9H7) to semantic binding (Adaptive L9H1) was not a random mutation; it was the direct result of an architectural constraint that made redundant syntactic copying prohibitively expensive in the loss landscape.

7.3 Implications for Safety and Oversight

The emergence of the "Load-Bearing Head" (Adaptive L9H1) has significant implications for model oversight. In the Control model, the diffuse distribution of functional importance makes it difficult to locate specific capabilities; removing any single head has negligible impact. In the Adaptive model, the concentration of semantic binding into specific, identifiable heads creates natural "intervention points."

For safety practitioners, this suggests that orthogonal regularization could facilitate **Targeted Monitoring**. If a model is trained with VSM, we can identify the specific heads responsible for high-level reasoning or semantic retrieval and attach probes or "circuit breakers" directly to those components, rather than monitoring the entire high-dimensional residual stream.

7.4 Limitations and Future Work

Scale: This study was conducted on “Nano-LLMs” (40M parameters). While rank collapse is theoretically more acute in small models, it remains to be seen if VSM provides similar benefits at the 7B+ scale, where the “Lottery Ticket” probability is higher.

Computational Overhead: The calculation of the Gram matrix scales quadratically with the number of heads ($O(H^2)$). While negligible for 8 heads, this could become a bottleneck for models with massive head counts (e.g., GPT-4 scale). Future work should explore “Local VSM” (enforcing orthogonality only within local groups of heads) or stochastic approximation methods.

Causality vs. Correlation: While we established a causal link between the *intervention* (VSM) and the *structure* (Orthogonality), the link between *structure* and *logic* remains inferential. Does orthogonality *cause* better logic, or does it simply remove the noise that prevents logic from emerging? Further research using causal scrubbing is required to fully map the circuit-level mechanism.

8 Conclusion

This research addressed “Rank Collapse,” a critical structural pathology that limits the expressive capacity of small-parameter language models. While Part I of Project Janus demonstrated that Vector Space Homeostasis (VSM) could improve logical coherence by 9.2%, this paper has provided the mechanistic evidence required to explain *why* that improvement occurs.

By subjecting the Janus v3 models to a rigorous “Glass Box” analysis using Sparse Autoencoders, topology metrics, and ablation studies, we have uncovered three fundamental structural shifts induced by orthogonal regularization:

1. **Subspace Disentanglement:** VSM successfully reduced inter-head correlation by **60.3%** in the deepest layers, physically separating the representation subspaces and preventing the model from collapsing into redundant feature learning.

2. **The Sparsity Crossover:** We identified a novel computational strategy where VSM-trained models exhibit **12.3% higher selectivity** (sparsity) in early layers and **25.5% higher expressivity** (density) in deep layers. This “Filter-then-Pack” hierarchy contrasts sharply with the uniform, noisy activation profile of standard models.

3. **Functional Specialization:** The emergence of “load-bearing” heads—specifically **Layer 9, Head 1**, which carried **2.54x** the functional importance of any control head—confirms that geometric constraints force components to assume distinct, non-redundant roles. Crucially, this specialization manifested as a qualitative shift from syntactic scaffolding to semantic binding.

These findings validate the “Super-Chinchilla” hypothesis: that the effective parameter count of a model is not fixed by its architecture, but is a function of its rank utilization. By enforcing architectural hygiene through VSM, we effectively “created” additional representational capacity, allowing a 40M parameter model to exhibit the structural complexity of a larger system.

Ultimately, this work establishes **Prescriptive Interpretability** as a viable paradigm for AI research. Rather than accepting the opaque, entangled structures that emerge from standard training, we have shown that it is possible to design objectives that actively shape the internal topology of the model. In an era dominated by scaling laws, VSM serves as a reminder that *how* we use parameters matters just as much as *how many* we have.

9 Reproducibility Statement

To facilitate the validation and extension of these findings, we have released the complete "Janus Mechanistic Workbench" as an open-source repository.

Artifacts Available:

- **Model Checkpoints:** Full PyTorch state dictionaries for both the `Janus-v3-Control` and `Janus-v3-Adaptive` (40M) models.
- **Telemetry Data:** The raw CSV files used to generate the tables in this paper, including `topology_stats.csv` (rank analysis), `sae_training_results.csv` (feature density), and `ablation_results.csv` (functional impact).
- **Codebase:**
  - `train.py`: The training script implementing the P-Controller and VSM loss.
  - `sae_runner.py`: The complete pipeline for harvesting activations and training Sparse Autoencoders.
  - `ablation.py`: The zero-mask ablation testing suite.
  - `topology.py`: The stable rank and cosine similarity analysis tools.

Experimental Context: All experiments were conducted on a single NVIDIA L4 GPU (24GB VRAM). The training budget was strictly capped at 4,005 steps per model. We have provided the random seeds and initialization distributions in the configuration files to ensure that the specific trajectory of the "Ignition" and "Homeostasis" phases can be replicated exactly.

References

Jonathan R. Belanger. Engineering efficient nano-llms via feature orthogonality and vector space homeostasis. *Exorobourii LLC Technical Report*, 2025. Project Janus Part I.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, et al. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. URL https://transformer-circuits.pub/2023/monosemantic-features/index.html.

Michael Cogswell, Faruk Ahmed, Ross Girshick, Larry Zitnick, and Dhruv Batra. De-covaring representations in neural networks. In *International Conference on Learning Representations*, 2016.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. *arXiv preprint arXiv:2309.08600*, 2023.

Yihe Dong, Jean-Baptiste Cordonnier, and Andreas Loukas. Attention is not all you need: Pure attention loses rank doubly exponentially with depth. In *International Conference on Machine Learning*, pages 2793–2803. PMLR, 2021.

Ronen Eldan and Yuanzhi Li. Tinystories: How small can language models be and still speak coherent english? *arXiv preprint arXiv:2305.07759*, 2023.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Jack Lasenby, Dawn Drain, Carol Chen, et al. Toy models of superposition. *Transformer Circuits Thread*, 2022. URL https://transformer-circuits.pub/2022/toy_model/index.html.

Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. *arXiv preprint arXiv:1803.03635*, 2018.

Suriya Gunasekar, Yi Zhang, J Jyothi, Sébastien Bubeck, Ronen Eldan, Yin Tat Lee, and Yuanzhi Li. Textbooks are all you need. *arXiv preprint arXiv:2306.11644*, 2023.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

Paul Michel, Omer Levy, and Graham Neubig. Are sixteen heads really better than one? In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Olivier Roy and Martin Vetterli. The effective rank: A measure of effective dimensionality. In *2007 15th European Signal Processing Conference*, pages 606–610. IEEE, 2007.

Noam Shazeer. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020.

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5797–5808, 2019.

Biao Zhang and Rico Sennrich. Root mean square layer normalization. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

A Experimental Hyperparameters

To ensure reproducibility, we detail the exact configuration of the "Janus v3" architectural chassis, the training hyperparameters, and the Sparse Autoencoder (SAE) settings used in this study.

Table 4: Model Architecture (Janus v3 Chassis)

Parameter	**Value**	**Description**
Parameters	$\approx 40,000,000$	Matches GPT-2 Small scale
Layers	12	Standard depth
Attention Heads	8	Standard width
Model Dimension (d_{model})	512	Embedding size
Head Dimension (d_{head})	64	d_{model}/N_{heads}
Context Window	512	Training context length
Vocabulary	50,257	GPT-2 Tokenizer
Position Embedding	RoPE	Rotary Positional Embeddings
Activation	SwiGLU	Gated Linear Unit (Scale 8/3)
Normalization	RMSNorm	Root Mean Square Layer Norm
Bias Terms	False	All biases disabled
Initialization	$\mathcal{N}(0, 0.02)$	Residual scaling $1/\sqrt{2L}$ applied

Table 5: Training Configuration

Parameter	**Value**	**Notes**
Optimizer	AdamW	Standard implementation
Learning Rate	6.29×10^{-4}	Peak LR
Weight Decay	1.34×10^{-4}	Regularization
Batch Size	16	Per device
Gradient Accumulation	4	Effective Batch Size = 64
Total Steps	4,005	$\approx$ 1 epoch of WikiText-103
Precision	`bfloat16`	Mixed precision training
Hardware	NVIDIA L4	24GB VRAM

Table 6: Sparse Autoencoder (SAE) Settings

Parameter	**Value**	**Description**
Input Dimension	64	Single Head Output (d_{head})
Expansion Factor	8×	Ratio of Hidden to Input
Hidden Dimension	512	Number of learned features
Activation	ReLU	Rectified Linear Unit
Training Data	10M Tokens	Harvested from WikiText-103
Loss Function	MSE + L1	Reconstruction + Sparsity

B The Homeostatic Control Algorithm

The core innovation of the Adaptive model is the **Homeostatic P-Controller**, which dynamically adjusts the orthogonality penalty (λ) based on the model's current state. This replaces static regularization schedules with a closed-loop feedback system.

Listing 1: Vector Space Homeostasis (VSM) Controller

```
class Homeostat:
    def __init__(self, target_sigma=0.0035, k_p=0.5, warmup_steps=1500):
        self.target = target_sigma   # Desired max correlation
        self.kp = k_p                # Proportional Gain
        self.warmup = warmup_steps   # Ignition Phase duration
        self.lambda_val = 0.0        # Current regularization strength

    def update(self, step, current_sigma_avg):
        """
        Calculates the new lambda based on the current step and
        measured head correlation (sigma).
        """
        # Phase 1: Ignition (Open Loop)
        # Linearly ramp pressure to allow initial circuit formation
        if step < self.warmup:
            self.lambda_val = 0.05 * (step / self.warmup)
            return self.lambda_val

        # Phase 2: Homeostasis (Closed Loop)
        # Apply P-Control to maintain target orthogonality
        if current_sigma_avg is None:
            return self.lambda_val

        # Error Calculation: Positive error means too much correlation
        error = current_sigma_avg - self.target

        # Proportional Update
        delta = self.kp * error
        self.lambda_val += delta

        # Safety Clamping
        self.lambda_val = max(0.0, min(0.25, self.lambda_val))

        return self.lambda_val
```

C Semantic Feature Atlas

This appendix provides a qualitative comparison of the features learned by the "Load-Bearing" heads identified in the Ablation Study (Section 6). We contrast the top activating features of the **Control Model (Layer 9, Head 7)** against the **Adaptive Model (Layer 9, Head 1)**.

Table 7: Control Model (L9H7) - Syntactic Scaffolding

Feature ID	**Token**	**Activation**	**Context / Interpretation**
224	"that"	2.32	Relative clause initiator
324	"has"	2.18	Auxiliary verb / possession
244	"of"	1.64	Prepositional linkage
012	"in"	1.21	Locative preposition
108	","	1.15	Clause separator

Table 8: Adaptive Model (L9H1) - Semantic Binding

Feature ID	**Token**	**Activation**	**Context / Interpretation**
362	"lazy"	1.62	Adjective (Behavioral attribute)
477	"fox"	0.94	Noun (Animal entity)
105	"complex"	0.88	Adjective (System property)
293	"brown"	0.72	Adjective (Visual attribute)
055	"physics"	0.68	Noun (Abstract domain)