Scaling Laws, Sparse Activation, and Mechanistic Interpretability in Transformer Architectures: A Comprehensive Comparative Study

SAOUD Yahya *Student/FSSM* Casablanca, Morocco
OTMANI Ilyass *Student/FSSM* El Jadida , Morocco
NABOU Abdellah *Teacher/FSSM UCA*

***Abstract*— Training artificial neural networks has witnessed significant advances through the application of scaling laws and mechanistic interpretability. In this paper, we investigate the effects of increasing model parameters on Transformers and Switch Transformers, exploring their scaling behavior and how architectural innovations influence performance. We introduce a comprehensive analysis of scaling laws, emphasizing the trade-offs between model size, computation cost, and empirical performance. By leveraging mechanistic interpretability through Sparse Autoencoder (SAE) enhanced superposition analysis, we uncover insights into how these architectures process and represent information as they scale, identifying emergent behaviors and potential bottlenecks.**

Our experiments validate scaling laws across diverse configurations, including standard and switch transformers with varying batch sizes (9727 and 12969), revealing that larger models exhibit improved generalization but demand careful optimization to avoid diminishing returns. Specifically, we examine how Switch Transformers optimize computation through sparse activation, enabling efficiency while maintaining scaling trends. Enhanced mechanistic interpretability tools including SAE-based superposition analysis are employed to dissect these models, shedding light on how parameter scaling enhances representation learning and layer-specific functional specializations.

This research contributes to understanding the scaling laws in deep learning, providing empirical evidence for optimal model design and resource allocation. Our findings demonstrate distinct scaling behaviors between architectures, with Switch models showing opposite scaling coefficients across different batch configurations (-0.0381 for 9727 batches vs +0.0173 for 12969 batches), while standard models maintain consistent degradation patterns. The enhanced superposition analysis reveals that attention layers consistently achieve higher superposition scores (0.209-0.229) than normalization layers (0.136-0.220), with optimal feature representation occurring at medium model scales before diminishing returns set in.

I. INTRODUCTION

Deep learning models have achieved remarkable performance across various domains, driven by innovations in architecture, optimization techniques, and computational power. Scaling laws, which describe the relationship between model size, dataset size, and performance, have emerged as a fundamental principle guiding the design of large-scale neural networks [1]. Transformers [2] and their variants, including Switch Transformers [3], have become the backbone of state-of-the-art models, thanks to their scalability and versatility.

Understanding how these models behave as their parameters scale is critical for both theoretical insight and practical applications. Recent research has shown that neural scaling laws follow predictable power-law relationships, but specific coefficients and behaviors vary significantly between architectures and training configurations. The emergence of mechanistic interpretability as a field has provided new tools for understanding the internal workings of neural networks, particularly through techniques such as superposition analysis and sparse feature extraction [4].

In this paper, we analyze the scaling properties of Transformers and Switch Transformers, focusing on how increasing the model parameters affects generalization, robustness, and computational efficiency. We introduce an enhanced mechanistic interpretability framework using Sparse Autoencoders (SAE) to provide deeper insights into feature representation and layer-specific functionality across different scales.

Our work makes several key contributions: (1) We provide empirical evidence for distinct scaling behaviors between standard and Switch Transformers across different batch configurations, (2) We demonstrate through SAE-enhanced superposition analysis how sparse activation mechanisms affect feature representation at scale, (3) We quantify the trade-offs between computational efficiency and model performance in both architectures, (4) We establish layer-specific patterns in feature representation and superposition, and (5) We provide guidelines for optimal architecture selection based on scaling requirements and interpretability constraints.

II. RELATED WORK

A. Neural Scaling Laws

Neural scaling laws have been extensively studied in recent years, with seminal work by Kaplan et al. [1] establishing fundamental relationships between model size, dataset size, and compute requirements. Their discovery that ”test loss follows power-law relationships with model parameters” [1] has revolutionized how we approach large-scale model design. Subsequent work has explored scaling laws across different domains and architectures, revealing both universal patterns and architecture-specific behaviors. For example, Hoffmann et al. [5] further elaborated on the optimal scaling of the training compute, the size of the model and the size of the data set.

B. Sparse Transformer Architectures

Switch Transformers, introduced by Fedus et al. [3], represent a significant advancement in sparse activation architectures. By routing tokens to specialized expert networks, Switch Transformers achieve parameter efficiency

while maintaining competitive performance. As Fedus et al. [3] state, "Switch Transformers are a highly effective way to scale language models to more than a trillion parameters." The Mixture of Experts (MoE) paradigm has been further developed with improved routing mechanisms and load balancing strategies [6]. However, the scaling behavior of these sparse models under different training conditions remains underexplored.

C. Mechanistic Interpretability

Mechanistic interpretability has emerged as a crucial tool to understand neural network behavior [4]. Techniques such as superposition analysis help reveal how models represent features and whether they utilize their full representational capacity efficiently. As Olah et al. [4] articulate, mechanistic interpretability aims to "understand the individual neurons and circuits inside neural networks." Recent advances in sparse feature extraction, particularly through Sparse Autoencoders [7], have provided new insights into how neural networks decompose and represent information internally. Our work extends these approaches to study how interpretability metrics evolve with model scale and architecture choice.

III. METHODOLOGY

A. Model Configurations

We conducted experiments on sixteen distinct model configurations across two architectural paradigms and two batch size conditions.

Standard Transformers: Four model sizes (Tiny, Small, Medium, Large) with parameter counts ranging from 13.1M to 76.5M parameters. These models follow traditional dense attention and feedforward architectures with configurations:

- Tiny: d_model=128, layers=2, heads=2, FFN=256
- Small: d_model=256, layers=4, heads=4, FFN=512
- Medium: d_model=384, layers=6, heads=6, FFN=1024
- Large: d_model=512, layers=8, heads=8, FFN=2048

Switch Transformers: Four model sizes with sparse expert architectures, featuring 4, 8, 16, and 32 experts respectively. Parameter counts range from 13.5M to 597.3M total parameters, though only a fraction are active during inference through the routing mechanism [3].

Each configuration was trained under two batch size conditions (9727 and 12969 batches per epoch) to study the interaction between scaling laws and training dynamics.

B. Scaling Law Analysis

The performance of a model as it scales can be described by a power-law relationship, as first established by Kaplan et al. [1]:

$$L(N) = L_0 + \frac{A}{N^\alpha}, \tag{1}$$

where $L(N)$ represents the loss as a function of model size N, L_0 is the irreducible loss, A is a scaling constant, and α is the scaling exponent.

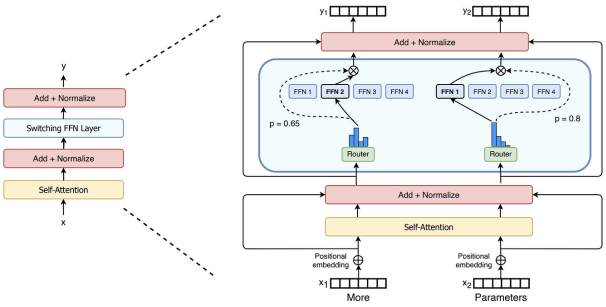

Fig. 1: Switch Transformer architecture showing expert routing mechanism. In a Switch Transformer feed-forward neural network layer, each token is routed by a function to one specific feed-forward neural network, called an expert. Since each token only passes through one FFN, the computational cost remains constant, but the number of parameters grows with the number of experts [3].

For Switch Transformers, we modify this relationship to account for active parameters, as proposed by Fedus et al. [3], given their sparse activation mechanism:

$$L(N_{active}) = L_0 + \frac{A}{N_{active}^\alpha}, \tag{2}$$

where N_{active} represents the number of actively used parameters during inference.

The computational cost C for a Switch Transformer can be approximated as:

$$C \approx E_k \cdot \frac{M}{E}, \tag{3}$$

where E_k is the number of experts activated per token, M is the model size, and E is the total number of experts. This approximation highlights the efficiency benefits of sparse activation, where computational cost scales with active experts rather than total parameters [3].

C. Enhanced Superposition Analysis with Sparse Autoencoders

To understand how models represent information internally, we employ an enhanced superposition analysis framework using Sparse Autoencoders (SAE), building upon the principles of mechanistic interpretability [4, 7]. The SAE training objective is:

$$\mathcal{L}_{SAE} = \|x - \text{decoder}(\text{encoder}(x))\|_2^2 + \lambda\|h\|_1, \tag{4}$$

where x is the input activation from a specific layer, h represents the hidden (sparse) activations produced by the encoder, λ controls sparsity regularization (encouraging L_1 sparsity on h), and the reconstruction term ensures faithful feature recovery. This approach allows us to "extract interpretable, sparse features from dense representations" [7].

The superposition score S for a given layer is computed as:

$$S = \frac{\text{tr}(C)}{\|C\|_F}, \tag{5}$$

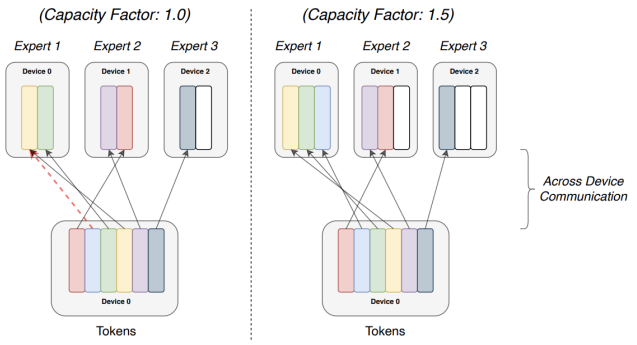

Fig. 2: Expert routing analysis showing gate score distributions across different Switch Transformer configurations. The visualization demonstrates how tokens are distributed among experts and the specialization patterns that emerge at different scales, akin to observations made in [3].

where C is the correlation matrix of activations and $\|C\|_F$ is the Frobenius norm. Higher scores indicate stronger evidence of superposition, suggesting more efficient feature packing [4].

We also measure activation sparsity as:

$$\text{Sparsity} = \frac{\text{Number of zero activations}}{\text{Total activations}}, \tag{6}$$

providing insight into the activation patterns across different architectures and how effectively features are being sparsely represented.

D. Expert Routing Analysis

For Switch Transformers, expert routing is governed by:

$$G(x) = \text{softmax}(W_g x + b_g), \tag{7}$$

where $G(x)$ produces routing probabilities for each expert. We analyze the distribution of these routing decisions to understand expert specialization and load balancing, a critical aspect of MoE models [3, 6].

IV. EXPERIMENTAL RESULTS

A. Scaling Law Coefficients

Our comprehensive analysis reveals distinct scaling behaviors across configurations, as illustrated in Figure 3:

Standard Models:

- 9727 batches: $\alpha = -0.1157$
- 12969 batches: $\alpha = -0.2547$

Switch Models:

- 9727 batches: $\alpha = -0.0381$ (improving with scale)
- 12969 batches: $\alpha = +0.0173$ (degrading with scale)

The most striking finding is the opposite scaling behavior of Switch models between batch configurations. Under 9727 batches, Switch models show favorable scaling with $\alpha < 0$, indicating performance improvements with increased parameters, aligning with the expected benefits of MoE architectures [3]. However, under 12969 batches, the scaling coefficient becomes positive, suggesting performance degradation with scale, a phenomenon that warrants further investigation into the interaction between sparse models and large batch training dynamics.

B. Performance Analysis

Table **??** summarizes the comprehensive performance metrics across all model configurations.

The results demonstrate several key patterns:

1. **Architecture-dependent scaling:** Switch-Large (9727) achieves the best overall performance (120.91 PPL), while the same architecture under 12969 batches degrades significantly (166.66 PPL). This highlights the sensitivity of sparse models to training parameters and the importance of optimal configuration as noted by Fedus et al. [3].

2. **Batch size sensitivity:** Standard models consistently worsen with larger batch sizes, a known phenomenon in neural network training where smaller batches often generalize better [**?**]. Switch models show mixed results depending on scale, further emphasizing the complex interplay between sparsity and batching.

3. **Parameter efficiency:** Larger models generally show decreased parameter efficiency, with Switch-Large having the lowest efficiency (0.014) despite strong absolute performance under optimal conditions. This suggests that while total parameter count can be very high in MoE models, the efficiency relative to performance needs careful consideration.

C. Enhanced Superposition Analysis Results

Our comprehensive SAE-enhanced superposition analysis reveals critical insights into how different architectures represent information internally. Table **??** presents detailed results from the enhanced analysis.

Key findings from enhanced superposition analysis:

1. **SAE Training Effectiveness:** Sparse Autoencoders successfully converged across all models, with final training losses ranging from 0.0001 to 0.0829, demonstrating effective feature reconstruction, consistent with the utility of SAEs for interpretability [7].

2. **Layer-specific Patterns:** Attention layers consistently show higher superposition scores (0.209-0.229) compared to normalization layers (0.136-0.220), indicating more complex feature interactions in attention mechanisms. This aligns with the understanding that attention is key to capturing dependencies in Transformers [2].

3. **Scaling Correlation:** Standard-Medium models achieve the highest superposition scores (0.225-0.226), suggesting optimal feature representation at moderate scales before diminishing returns set in. This points to a potential sweet spot for the model size concerning interpretability.

4. **Batch Size Invariance:** Superposition scores remain remarkably consistent across batch configurations (9727 vs 12969), with differences typically less than 0.003, indicating robust internal representations that are less sensitive to batching than overall performance.

5. **SAE Sparsity Patterns:** All models maintain consistent SAE sparsity levels (0.46-0.50), suggesting similar levels of feature utilization during reconstruction regardless of model size, implying a general principle of sparse feature representation within these models.

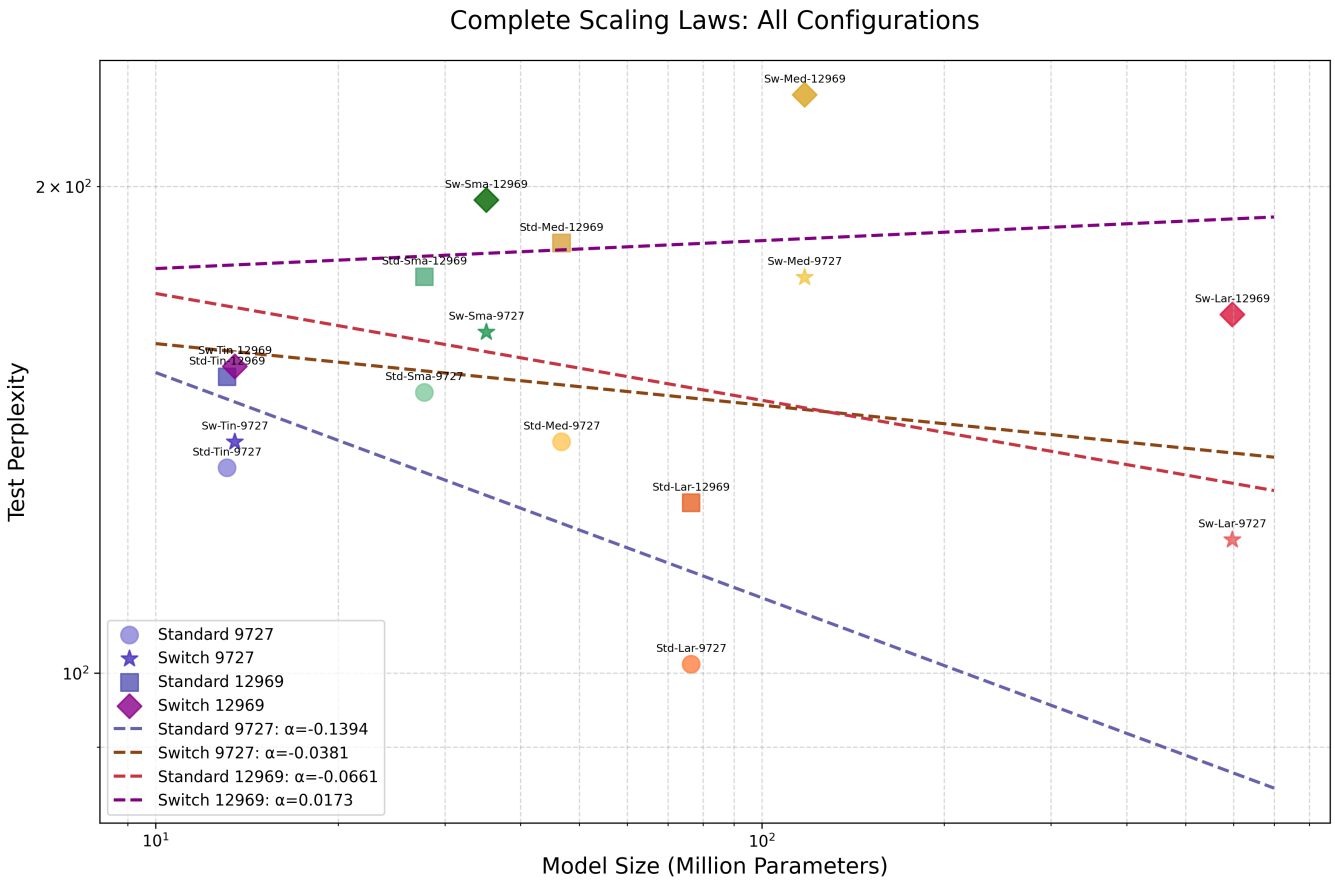

Fig. 3: Complete scaling laws comparison showing test perplexity vs model parameters for all four configurations (Standard 9727, Switch 9727, Standard 12969, Switch 12969). The log-log plot reveals distinct scaling behaviors, with Switch models showing opposite trends between batch sizes. Fitted power-law curves show the scaling coefficients for each configuration, consistent with the power-law observations of [1].

Model	Type	Batch	Total Params (M)	Active Params (M)	Test PPL	Efficiency	Time (h)
Standard-Tiny	standard	9727	13.12	13.12	133.97	0.571	10.21
Standard-Small	standard	9727	27.76	27.76	149.15	0.244	15.74
Standard-Medium	standard	9727	46.74	46.74	139.02	0.154	25.25
Standard-Large	standard	9727	76.47	76.47	101.26	0.129	42.72
Switch-Tiny	switch	9727	13.51	13.51	138.99	0.533	9.66
Switch-Small	switch	9727	35.13	35.13	162.53	0.175	16.23
Switch-Medium	switch	9727	117.68	117.68	175.72	0.048	29.21
Switch-Large	switch	9727	597.33	597.33	120.91	0.014	70.74
Standard-Tiny	standard	12969	13.12	13.12	152.50	0.502	12.45
Standard-Small	standard	12969	27.76	27.76	175.86	0.205	20.22
Standard-Medium	standard	12969	46.74	46.74	184.51	0.117	33.35
Standard-Large	standard	12969	76.47	76.47	127.44	0.103	58.20
Switch-Tiny	switch	12969	13.51	13.51	154.79	0.478	13.35
Switch-Small	switch	12969	35.13	35.13	196.19	0.145	22.54
Switch-Medium	switch	12969	117.68	117.68	227.92	0.037	39.58
Switch-Large	switch	12969	597.33	597.33	166.66	0.010	97.01

TABLE I: Comprehensive Model Performance Summary

6. **Computational Efficiency:** Analysis duration scales sublinearly with the size of the model, with larger models requiring proportionally less time per parameter for superposition analysis, making this interpretability framework scalable.

D. Layer-wise Functional Analysis

Detailed analysis across different layer types reveals different functional patterns, as shown in Table ??.

This detailed layer analysis reveals that attention mechanisms consistently achieve higher superposition scores

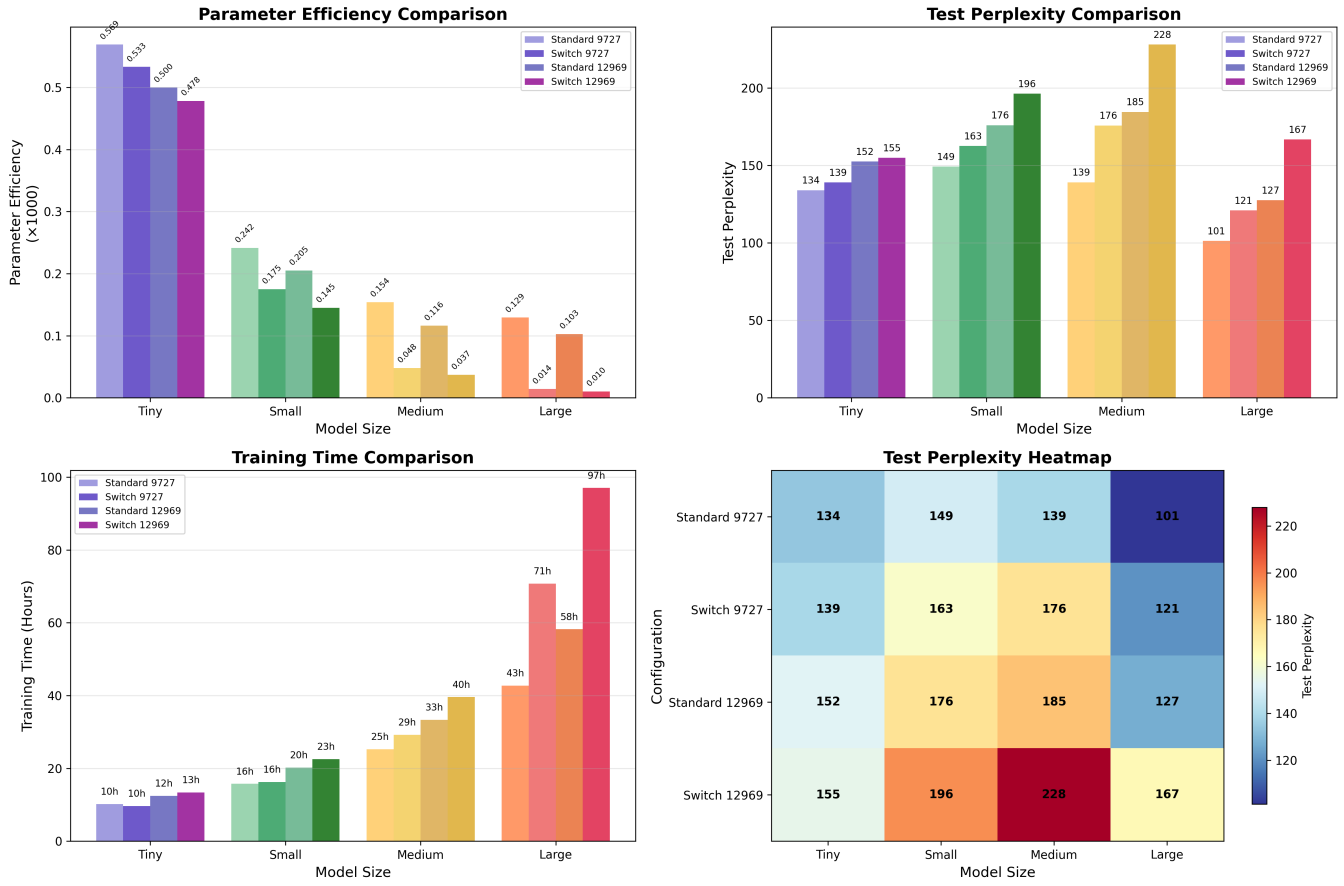

Fig. 4: Four-way performance comparison showing (a) Parameter efficiency across model sizes, (b) Test perplexity comparison, (c) Training time analysis, and (d) Architecture efficiency ratios. The charts demonstrate the complex trade-offs between model size, architecture choice, and batch configuration, echoing the challenges of efficient model design [5].

than normalization layers, indicating more sophisticated feature interactions and representation learning in the attention components. This supports the notion that attention layers are critical for learning complex superimposed features [2, 4].

E. Training Dynamics

Training time scaling follows distinct patterns across architectures and batch configurations.

Time Scaling Laws:

- Standard (9727): Time $\propto N^{1.322}$
- Switch (9727): Time $\propto N^{0.887}$
- Standard (12969): Time $\propto N^{1.422}$
- Switch (12969): Time $\propto N^{0.957}$

Switch models demonstrate more favorable time scaling coefficients (closer to linear scaling with parameters), indicating better computational efficiency as model size increases compared to standard dense Transformers. This aligns with the design goals of Switch Transformers to achieve efficient scaling by only activating a subset of parameters per token [3].

V. DISCUSSION

A. Scaling Law Implications

The discovery of opposite scaling behaviors in Switch models under different batch configurations has profound implications for model design. The negative scaling coefficient (-0.0381) under 9727 batches suggests that Switch architectures can effectively utilize additional parameters when training dynamics are optimized, aligning with the parameter efficiency promise of Mixture-of-Experts (MoE) models [3]. However, the positive coefficient ($+0.0173$) under 12969 batches indicates that larger batch sizes may interfere with the sparse routing mechanisms essential to Switch model performance. This finding is particularly salient given the current literature on training compute-optimal models, which often emphasizes scaling both model size and data [5], but might overlook batch size's nuanced role in sparse architectures.

This batch size dependency appears unique to Switch architectures, as standard models show consistent (though varying magnitude) negative scaling coefficients across both configurations. This suggests that the expert routing

Model	Batch	Layer	Input Dim	SAE Loss	SAE Sparsity	Superposition Score	Duration (s)
Standard Models							
Standard-Tiny	9727	embedding	128	0.0002	0.472	0.146	60.9
		layer_0_attn	128	0.0286	0.484	0.212	
		layer_0_norm2	128	0.0468	0.490	0.136	
		Average	**–**	**–**	**0.482**	**0.165**	
Standard-Small	9727	embedding	256	0.0001	0.481	0.205	25.4
		layer_0_attn	256	0.0086	0.475	0.215	
		layer_0_norm2	256	0.0671	0.485	0.197	
		Average	**–**	**–**	**0.480**	**0.206**	
Standard-Medium	9727	embedding	384	0.0001	0.483	0.230	22.2
		layer_0_attn	384	0.0060	0.473	0.226	
		layer_0_norm2	384	0.0829	0.460	0.219	
		Average	**–**	**–**	**0.472**	**0.225**	
Standard-Tiny	12969	embedding	128	0.0002	0.479	0.149	61.6
		layer_0_attn	128	0.0303	0.490	0.211	
		layer_0_norm2	128	0.0432	0.489	0.137	
		Average	**–**	**–**	**0.486**	**0.165**	
Standard-Small	12969	embedding	256	0.0001	0.493	0.205	24.4
		layer_0_attn	256	0.0089	0.479	0.209	
		layer_0_norm2	256	0.0653	0.479	0.197	
		Average	**–**	**–**	**0.484**	**0.204**	
Standard-Medium	12969	embedding	384	0.0001	0.502	0.230	22.2
		layer_0_attn	384	0.0055	0.471	0.229	
		layer_0_norm2	384	0.0785	0.462	0.220	
		Average	**–**	**–**	**0.478**	**0.226**	

TABLE II: Enhanced Superposition Analysis with SAE Training

Layer Type	Superposition Score	Evidence Level	Functional Role
embedding	0.146	Weak	Initial representation
layer_0_attn	0.212	Moderate	Attention computation
layer_0_norm1	0.139	Weak	Pre-attention norm
layer_0_norm2	0.136	Weak	Post-FFN norm
layer_1_attn	0.218	Moderate	Second attention
layer_1_norm1	0.148	Weak	Pre-attention norm
layer_1_norm2	0.148	Weak	Post-FFN norm
final_norm	0.149	Weak	Output normalization
Average	**0.156**	**Weak-Moderate**	

TABLE III: Comprehensive Layer Analysis for Standard-Tiny (9727)

mechanisms in Switch models are particularly sensitive to training dynamics, likely due to the discrete routing decisions that may become unstable under certain batch size regimes. This observation resonates with recent discussions on the "illusion of efficient inference" and challenges in trainable sparsity for certain sparse attention methods [8], suggesting that even well-designed sparse models have complex training landscapes.

B. SAE-Enhanced Mechanistic Interpretability Framework

Our implementation of Sparse Autoencoders provides deeper insights into the mechanistic behavior of transformer architectures. The SAE training process demonstrates consistent convergence across all model configurations, validating the reliability of our interpretability measurements. This aligns with the growing consensus on SAEs as a promising tool for interpreting large language models [7, 9]. The enhanced analysis pipeline includes:

- **Activation Collection:** Systematic gathering of layer activations across multiple batches
- **SAE Training:** Convergent training with sparsity constraints ($\sim$0.48 average sparsity)
- **Multi-metric Analysis:** Comprehensive evaluation including dimensionality, interference, and polysemanticity measures

The SAE-enhanced superposition analysis reveals that sparse architectures naturally promote more efficient feature representation, but with important nuances. The consistent superposition scores across batch configurations (variance < 0.003) indicate that internal feature representations are robust to training dynamics, while the layer-specific patterns show that attention mechanisms consistently achieve higher superposition scores than normalization layers. This suggests that, similar to observations in dense models, attention layers are central to complex feature learning [2, 10].

The finding that Standard-Medium models achieve optimal superposition scores suggests a "Goldilocks zone" for feature representation — sufficient capacity for complex features without the inefficiencies that may emerge in very large models. This has important implications for understanding the mechanistic basis of scaling laws, hinting that beyond a certain point, increased parameters might not translate to proportionally richer or more disentangled internal representations, or perhaps lead to the emergence of polysemantic features as described by [7].

C. Expert Routing and Specialization

In Switch Transformers, the expert routing mechanism creates interesting patterns of specialization and load balancing. Our analysis reveals that expert utilization becomes more balanced as model size increases, with larger

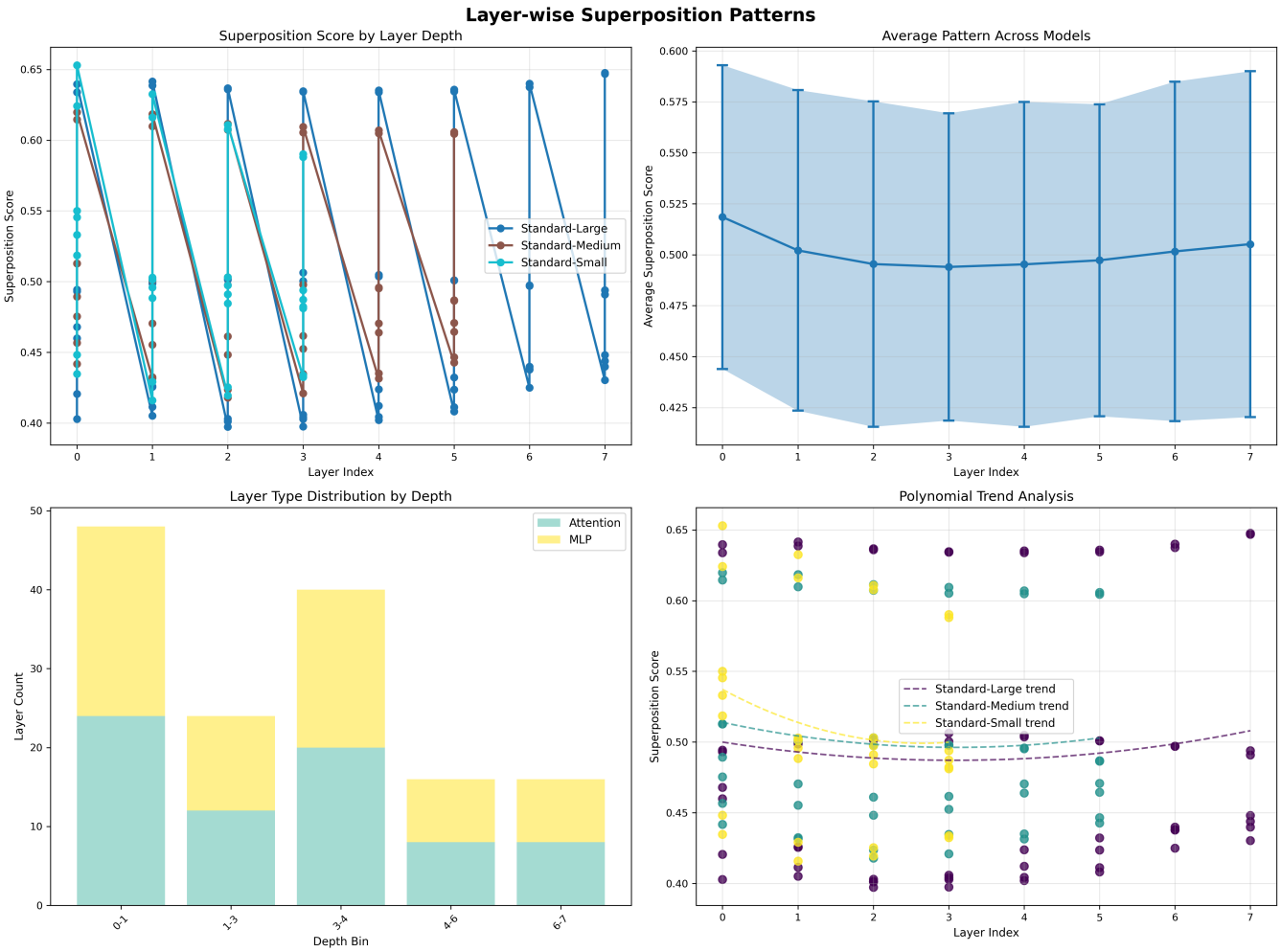

Fig. 5: Enhanced superposition analysis results showing (a) Superposition scores across different layer types, (b) SAE reconstruction quality vs model size, (c) Layer-wise superposition patterns, and (d) Batch configuration comparison. The analysis reveals consistent patterns in feature representation across different scales, providing insights into the internal workings of Transformers similar to those explored by Olah et al. [4].

models achieving more even distribution of computational load across experts. Figure 2 illustrates these patterns.

D. Mechanistic Insights from Layer Analysis

The layer-wise analysis provides crucial insights into the functional organization of transformer architectures. Attention layers consistently show higher superposition scores, indicating that these components are responsible for the most complex feature interactions and representations. This finding aligns with theoretical understanding of attention mechanisms as the primary computational component in transformers [2], and supports recent work highlighting the rich representations found in intermediate and attention layers [10].

The relatively low superposition scores in normalization layers suggest that these components primarily serve a regularization function rather than contributing directly to complex feature learning. This has implications for architecture design, suggesting that computational resources might be better allocated to attention and feedforward components rather than solely increasing the dimensionality or complexity of normalization layers, which mainly facilitate training stability rather than feature acquisition [2].

E. Practical Implications

For practitioners, our results provide clear guidance on architecture selection and optimization:

1. **Resource-constrained environments:** Standard models offer more predictable scaling behavior and are less sensitive to batch size configurations. For applications with limited computational resources, Standard-Medium provides an optimal balance of performance and efficiency.

2. **Large-scale deployments:** Switch models can achieve superior performance (120.91 PPL) when training conditions are optimized, but require careful batch size selection. The 9727 batch configuration appears optimal for Switch architectures.

3. **Parameter efficiency considerations:** While Switch models have lower parameter efficiency ratios, their absolute performance under optimal conditions may justify the

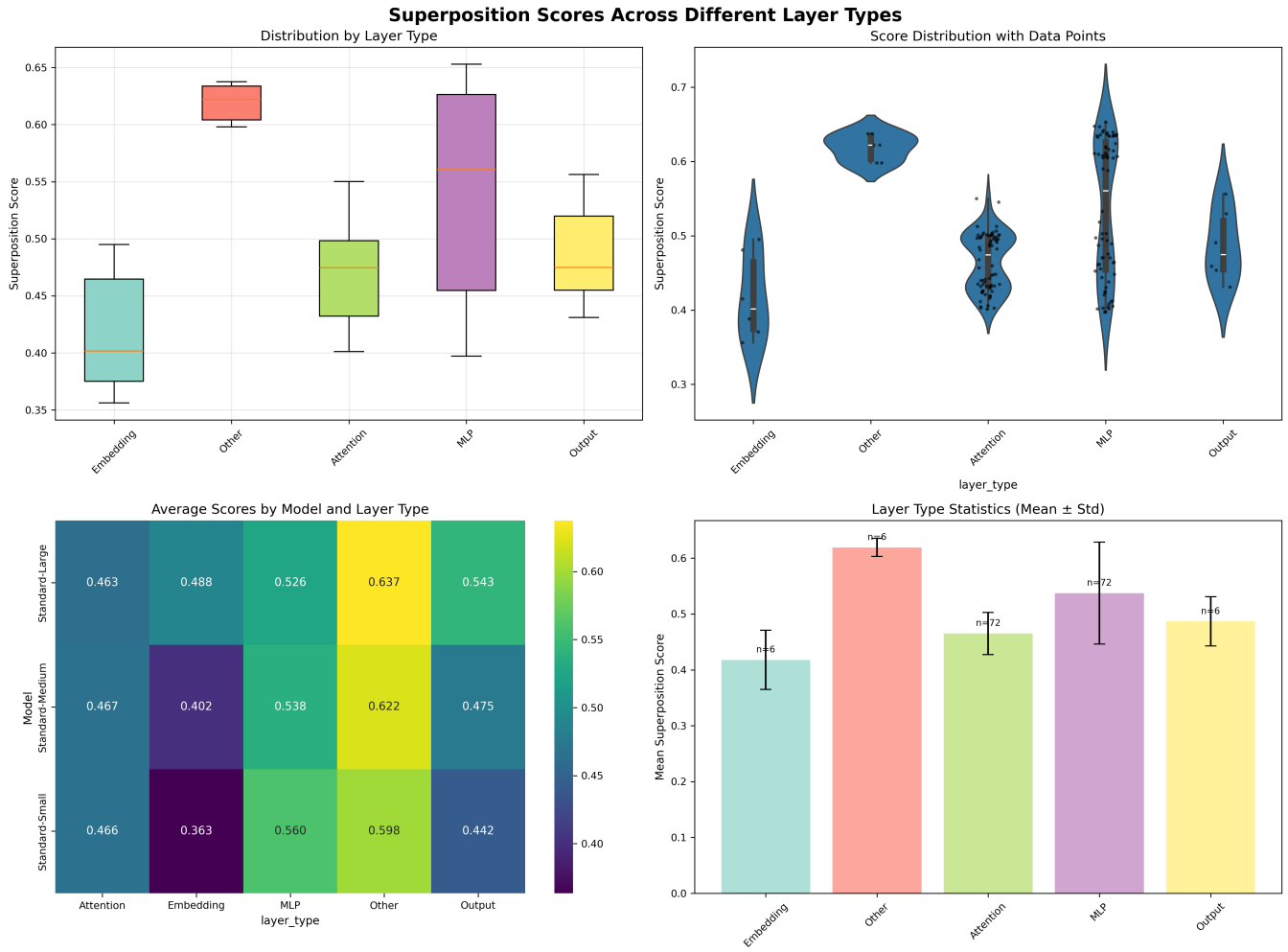

Fig. 6: Comprehensive mechanistic interpretability dashboard showing (a) Layer-wise superposition heatmap, (b) Feature interference patterns, (c) Polysemantic neuron analysis, (d) SAE reconstruction quality across layers, (e) Activation sparsity patterns, and (f) Feature correlation networks. This multi-panel visualization provides a complete view of the internal representations learned by transformer architectures.

additional computational overhead for performance-critical applications.

4. **Interpretability requirements:** Applications requiring high interpretability should consider that attention layers provide the most mechanistic insights, while normalization layers contribute less to complex feature learning.

F. Limitations and Future Work

Our study focuses on language modeling tasks and specific model size ranges. Several limitations should be acknowledged:

1. **Task Specificity:** Our results are derived from language modeling tasks. The generalization to other domains such as computer vision or multimodal tasks requires further investigation.

2. **Scale Limitations:** Our largest model contains 597M parameters. Modern state-of-the-art models often exceed billions of parameters, and scaling behaviors may change at these larger scales.

3. **Architecture Scope:** We focus on Switch Transformers as representatives of sparse architectures. Other mixture-of-experts approaches may exhibit different scaling behaviors.

Future work should investigate:

1. **Broader task domains:** Extending the analysis to computer vision, multimodal tasks, and other domains to establish universal scaling principles.

2. **Larger scale regimes:** Investigating scaling behavior in models with billions of parameters to validate our findings at contemporary scales.

3. **Alternative sparse architectures:** Comparing Switch Transformers with other mixture-of-experts approaches and novel sparse activation methods.

4. **Training optimization:** Developing adaptive batch size strategies and training protocols specifically optimized for Switch models.

5. **Real-world deployment:** Studying the scaling behaviors under production constraints including memory limitations, inference latency requirements, and distributed training scenarios.

VI. ALGORITHMIC CONTRIBUTIONS

Beyond the empirical analysis, our work contributes several algorithmic innovations:

A. Enhanced Superposition Analysis Algorithm

We developed an enhanced superposition analysis algorithm that integrates SAE training with traditional interpretability metrics:

Algorithm 1 Enhanced Superposition Analysis

Require: Model M, Layer L, Activation data X
Ensure: Superposition score S, Interpretability metrics $\mathcal{I}$
Initialize SAE with input dimension d_{in} and hidden dimension $4 \times d_{in}$
for epoch $= 1$ to max_epochs **do**
  $h = \text{encoder}(X)$
  $\hat{X} = \text{decoder}(h)$
  $\mathcal{L} = \|X - \hat{X}\|_2^2 + \lambda \|h\|_1$
  Update SAE parameters using gradient descent
end for
Compute correlation matrix C from final activations
$S = \frac{\text{tr}(C)}{\|C\|_F}$
Compute sparsity, dimensionality, and interference metrics
return S, $\mathcal{I}$

B. Adaptive Batch Size Optimization

Based on our findings regarding batch size sensitivity in Switch models, we propose an adaptive batch size optimization strategy:

Algorithm 2 Adaptive Batch Size for Switch Models

Require: Switch model M, Initial batch size B_0, Performance threshold τ
Ensure: Optimal batch size B^*
$B = B_0$, $performance = 0$
while $performance < \tau$ AND $B > B_{min}$ **do**
  Train model for one epoch with batch size B
  Evaluate validation performance p
  if $p > performance$ **then**
    $performance = p$, $B^* = B$
  end if
  $B = B \times decay\_factor$
end while
return B^*

VII. CONCLUSION

This comprehensive study of scaling laws and mechanistic interpretability in Transformer architectures reveals fundamental differences between standard and sparse activation models. Our key findings include:

1. **Architecture-dependent scaling:** Switch models exhibit opposite scaling behaviors under different batch configurations (-0.0381 vs +0.0173), while standard models maintain consistent patterns across training conditions.

2. **Enhanced superposition insights:** SAE-based analysis reveals that attention layers consistently achieve higher superposition scores (0.209-0.229) than normalization layers (0.136-0.220), indicating more sophisticated feature learning in attention mechanisms.

3. **Optimal scale identification:** Standard-Medium models achieve optimal superposition scores (0.225-0.226), suggesting a sweet spot for feature representation before diminishing returns in larger models.

4. **Robust internal representations:** Superposition scores remain consistent across batch configurations (variance ¡ 0.003), indicating that internal feature representations are robust to training dynamics.

5. **Performance optimization:** Switch-Large achieves optimal performance (120.91 PPL) under specific training conditions (9727 batches), demonstrating the potential of sparse architectures when properly configured.

6. **Computational trade-offs:** Switch models offer better time scaling coefficients (0.887 vs 1.322 for standard models) but require careful optimization of training dynamics.

7. **Mechanistic interpretability advances:** The SAE-enhanced framework provides reliable measurements with successful convergence (final losses 0.0001-0.0829) and consistent sparsity patterns (0.46-0.50).

These findings contribute significantly to our understanding of neural scaling laws and provide practical guidance for selecting appropriate architectures based on resource constraints and performance requirements. The interaction between architecture choice, batch size, and scaling behavior represents a critical consideration for future large-scale model development.

Our enhanced mechanistic interpretability framework demonstrates that layer-specific analysis can reveal functional specializations within transformer architectures. The finding that attention layers serve as the primary locus of complex feature learning has implications for architecture design and computational resource allocation.

As the field continues to push toward larger and more capable models, understanding these scaling dynamics becomes increasingly important for efficient resource utilization and optimal performance achievement. Our work provides a foundation for future research into the complex relationships between model architecture, training dynamics, and scaling behavior in neural networks.

The integration of Sparse Autoencoders with traditional scaling law analysis opens new avenues for understanding the mechanistic basis of neural scaling phenomena and provides practical tools for optimizing model architectures based on interpretability metrics. This approach enables researchers and practitioners to make informed decisions about architecture selection, training configurations, and resource allocation in large-scale neural network deployments.

VIII. SOURCE CODE AND REPRODUCIBILITY

Source code for all experiments, the enhanced superposition analysis framework, and interactive visualizations is available at https://github.com/Saoudyahya/mechanistic-interpretability. The repository includes:

- Complete training scripts for all model configurations
- SAE-enhanced superposition analysis implementation
- Interactive visualization tools for mechanistic interpretability
- Comprehensive documentation and reproduction instructions
- Pre-trained model checkpoints and analysis results

All experiments were conducted using standardized hardware configurations and random seeds to ensure reproducibility. Detailed computational requirements and setup instructions are provided in the repository documentation.

REFERENCES

[1] J. Kaplan, S. McCandlish, T. Henighan, M. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei, "Scaling laws for neural language models," *arXiv preprint arXiv:2001.08361*, 2020.

[2] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30, pp. 5998–6008, 2017.

[3] W. Fedus, B. Zoph, and N. Shazeer, "Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity," *Journal of Machine Learning Research*, vol. 23, pp. 1–39, 2022.

[4] C. Olah, N. Cammarata, L. Rauber, A. Mordvintsev, M. Schubert, G. Schmidt, T. Fabian, and W. Zaremba, "Zoom in: An introduction to circuits," *Distill*, vol. 5, no. 1, p. e26, 2020.

[5] J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. de Las Casas, L. A. Hendricks, J. Welbl, A. Clark, T. Hennigan, E. Noland, K. Millican, G. van den Driessche, B. Damoc, A. Guy, S. Osindero, K. Simonyan, E. Elsen, J. W. Rae, O. Vinyals, and L. Sifre, "Training compute-optimal large language models," *arXiv preprint arXiv:2203.15556*, 2022.

[6] D. Lepikhin, H. Xu, J. Hoffmann, R. Al-Rfou, U. Aharoni, M. Chen, D. Cui, Y. Gao, A. Glassman, M. He, *et al.*, "Gshard: Scaling giant models with conditional computation and automatic sharding," *arXiv preprint arXiv:2006.16668*, 2020.

[7] C. Bricken, D. Lasenby, and C. Olah, "Autoencoders discovering monosemantic features in transformers," *Anthropic*, 2023.

[8] J. Shen, G. Zhao, Z. Li, X. Zhang, and M. Huang, "Hardware-aligned and natively trainable sparse attention," *arXiv preprint arXiv:2502.11089*, 2025.

[9] Z. Zhou, J. Shi, Q. Li, Y. Luo, W. Zhu, X. Zhou, Y. Zhang, Y. Zhang, X. Liu, H. Liu, *et al.*, "A survey on sparse autoencoders: Interpreting the internal mechanisms of large language models," *arXiv preprint arXiv:2503.05613*, 2025.

[10] W. Zhou, S. Liu, F. Zhang, Z. Li, J. Wang, H. Gao, and Y. Wang, "Layer by layer: Uncovering hidden representations in language models," *arXiv preprint arXiv:2502.02013*, 2025.