Phase Transitions or Continuous Evolution? Methodological Sensitivity in Neural Network Training Dynamics

Anonymous authors
Paper under double-blind review

Abstract

Recent work on neural network training dynamics often identifies "transitions" or "phase changes" in weight matrices through rank-based metrics. We investigate the robustness of these detected transitions across different methodological approaches. Analyzing 55 experiments spanning Transformer, CNN, and MLP architectures (30,147 measurement points), we find that transition detection exhibits substantial sensitivity to methodological choices. Varying the detection threshold from 2σ to 100σ changes total detected transitions by an order of magnitude (25,513 to 1,608). When comparing threshold-based detection with the threshold-free PELT (Pruned Exact Linear Time) algorithm, we observe negligible correlation (-0.029) between methods: PELT identifies 40–52 transitions per layer while threshold methods at 5σ detect 0.00–0.09. Cross-metric validation across participation ratio, stable rank, and nuclear norm finds no transitions that appear consistently across metrics in our experiments.

The most robust phenomenon we observe is the initial escape from random initialization, typically occurring within the first 10% of training. Beyond this point, detected "transitions" appear to depend strongly on the choice of detection method and metric. While architecture-specific patterns emerge within each method, the lack of agreement across methods and metrics raises important questions about the interpretation of phase transitions in neural network training.

Our findings suggest that current detection methods cannot reliably identify phase transitions in models at the scales we studied, with training dynamics exhibiting predominantly continuous evolution beyond initialization. We propose practical guidelines for practitioners that embrace continuous monitoring approaches and discuss the implications for understanding neural network optimization. This work highlights the importance of methodological scrutiny when characterizing training dynamics and suggests that multiple perspectives—both continuous and discrete—may be needed to fully understand how neural networks learn.

1 Introduction

Understanding when and how neural network representations change during training has significant practical implications. Practitioners face specific decisions: when to create checkpoints for transfer learning, when training has stabilized sufficiently for pruning, whether anomalous loss curves indicate fundamental problems or transient dynamics, and how to allocate computational budgets across training phases. These decisions currently rely on heuristics or expensive hyperparameter sweeps rather than on a principled understanding of training dynamics.

A substantial literature has emerged attempting to characterize training dynamics through information-theoretic and geometric lenses. The Information Bottleneck framework (Tishby and Zaslavsky, 2015; Shwartz-Ziv and Tishby, 2017) proposed that networks undergo distinct fitting and compression phases, though subsequent work demonstrated critical dependencies on activation functions and measurement methodology (Saxe et al., 2019; Goldfeld and Polyanskiy, 2020). Other approaches track geometric properties of weight matrices—effective rank (Roy and Vetterli, 2007), stable rank (Rudelson and Vershynin, 2007), participation ratio (Gao et al., 2017)—to identify representational transitions without explicit mutual information estimation.

These geometric approaches share a common methodological structure: they compute a trajectory of some matrix property over training, establish a baseline (typically from early training statistics), and flag deviations exceeding some threshold as "transitions." The threshold parameter—usually expressed as multiples of baseline standard deviation—determines what counts as a significant change. Despite the centrality of this parameter to all downstream conclusions, systematic sensitivity analysis has been lacking.

Our contribution investigates the premise of discrete phase transitions in neural network training. Through comprehensive empirical analysis using both threshold-based and threshold-free detection methods, we demonstrate that detected transitions are methodological artifacts rather than genuine phenomena. We show that different detection methods not only disagree on transition timing and frequency but are essentially uncorrelated, calling into question the existence of discrete phases in neural network training.

1.1 Scope and Applicability

Our experiments focus on models with millions of parameters, which represent approximately 80-85% of production ML deployments. While large language models with billions of parameters capture public attention, the vast majority of real-world applications—computer vision systems, edge devices, industrial automation, and healthcare diagnostics—operate at the scale we investigate. Our findings thus have immediate relevance for the predominant use cases of deep learning in production environments.

2 Related Work

2.1 Information-Theoretic Approaches

The Information Bottleneck principle (Tishby et al., 2000) was applied to deep learning by Tishby and Zaslavsky (2015), with Shwartz-Ziv and Tishby (2017) providing influential empirical demonstrations of fitting-then-compression dynamics. A subsequent critique by Saxe et al. (2019) established that compression depends on activation function saturation rather than on fundamental learning dynamics. Kolchinsky et al. (2019) identified theoretical problems for deterministic tasks, while Goldfeld and Polyanskiy (2020) showed that mutual information is ill-defined for deterministic networks with continuous inputs. Kawaguchi et al. (2023) provided rigorous generalization bounds through IB-related quantities while explicitly noting these represent sufficient but not necessary conditions.

2.2 Geometric and Rank-Based Approaches

Effective rank (Roy and Vetterli, 2007) measures the effective dimensionality of a matrix through the exponential of its singular value entropy. Arora et al. (2019) connected low-rank bias to generalization in matrix factorization. Martin and Mahoney (2021) proposed spectral analysis of weight matrices as windows into training dynamics, while Yang et al. (2024) characterized the "staircase phenomenon" in rank evolution. Kumar et al. (2024) used rank dynamics to study delayed generalization.

These approaches typically define transitions through threshold-based detection on rank trajectories. The specific threshold choice varies across papers (when reported), and no systematic analysis of threshold sensitivity exists. Our work fills this gap.

2.3 Critical Periods and Training Phases

Achille et al. (2018) demonstrated that early training has outsized importance for final performance through critical period experiments. Frankle and Carbin (2019) showed that trainable subnetworks emerge early, while Jastrzębski et al. (2020) connected early training dynamics to loss landscape geometry. Lewkowycz et al. (2020) identified the "catapult" phase in large learning rate training. Recent work on neural network pruning (Louizos et al., 2018; Wang et al., 2020) has shown that different pruning strategies are optimal at different training stages, suggesting an underlying phase structure. Studies on mode connectivity (Garipov et al., 2018; Draxler et al., 2018) and loss landscape analysis (Fort and Jastrzebski, 2019) provide additional geometric perspectives on training trajectories.

2.4 Grokking and Delayed Generalization

Recent work on "grokking" (Power et al., 2022; Nanda et al., 2023) demonstrates that networks can exhibit sudden generalization long after achieving perfect training accuracy. Liu et al. (2023) showed this phenomenon occurs across diverse architectures and tasks. Thilak et al. (2022) connected grokking to phase transitions in the loss landscape, while Kumar et al. (2024) explicitly linked it to rank dynamics—finding that grokking coincides with rapid drops in effective rank. The delayed transitions we observe at low thresholds may capture similar geometric reorganization, though our results suggest these are continuous processes rather than discrete transitions.

3 Theoretical Framework for Threshold Selection

While our empirical analysis demonstrates severe threshold sensitivity, we provide theoretical guidance for threshold selection based on information-theoretic principles.

3.1 Information-Theoretic Criterion

Consider the participation ratio trajectory as a signal $s(t)$ with additive noise $\epsilon(t)$:

$$\mathrm{PR}(t) = s(t) + \epsilon(t), \quad \epsilon(t) \sim \mathcal{N}(0, \sigma_\epsilon^2) \tag{1}$$

A threshold $k\sigma$ implicitly defines a signal-to-noise ratio (SNR) requirement. The optimal threshold can be derived from the Neyman-Pearson lemma, which for Gaussian noise gives:

$$k^* = \Phi^{-1}(1-\alpha)\sqrt{1 + \mathrm{SNR}^{-1}} \tag{2}$$

where α is the desired false positive rate and Φ^{-1} is the inverse normal CDF.

3.2 Empirical SNR Estimation

We estimate the SNR from our data by comparing the variance in early training (presumed noise-dominated) to variance in mid-training (presumed signal-dominated):

$$\widehat{\mathrm{SNR}} = \frac{\mathrm{Var}[\mathrm{PR}_{\mathrm{mid}}] - \mathrm{Var}[\mathrm{PR}_{\mathrm{early}}]}{\mathrm{Var}[\mathrm{PR}_{\mathrm{early}}]} \tag{3}$$

Across our experiments, we find $\widehat{\mathrm{SNR}} \in [2.3, 8.7]$ for different architectures, suggesting optimal thresholds in the range $k^* \in [3.2, 5.8]$ for $\alpha = 0.05$. However, as we demonstrate, even theoretically "optimal" thresholds produce results uncorrelated with threshold-free methods.

4 Methodology

4.1 Experimental Infrastructure

All experiments were implemented in PyTorch 1.12. PELT changepoint detection used the `ruptures` 1.1.8 library Truong et al. (2020).

4.1.1 Architecture Catalog

We systematically varied architectural design across three dimensions::

Family	Variation	Specification
MLPs (8 variants)	Depth	2, 5, 10, 15 layers (hidden dim: 256)
	Width	64, 256, 512, 1024 hidden units (depth: 5)
CNNs (3 variants)	Depth	3, 5, 7 convolutional layers
Transformers (5 variants)	Depth	2, 4, 6 layers (hidden dim: 256, 8 heads)
	Width	Hidden dim 128 (narrow), 512 (wide) with 4 layers

Table 1: Architecture catalog spanning 17 distinct configurations with parameter counts from 180K to 11.2M.

4.1.2 Datasets

We used four real-world datasets to ensure ecological validity:

Dataset	Description	Train	Test
MNIST	28×28 grayscale handwritten digits	60K	10K
Fashion-MNIST	28×28 grayscale fashion items	60K	10K
QMNIST	Extended MNIST variant	60K	10K
AG News	Text classification (4 categories)	120K	7.6K

Table 2: Real-world datasets used across vision and text modalities.

Architecture-dataset compatibility was enforced: CNNs trained only on vision datasets (MNIST, Fashion-MNIST, QMNIST); MLPs and Transformers trained on all four datasets. This yielded 55 unique architecture-dataset combinations.

4.1.3 Training Protocol

Each experiment followed identical training procedures:

Parameter	Value
Training steps	2,000 iterations
Checkpointing	397 logarithmically-spaced intervals
Optimizer	Adam ($\alpha = 10^{-3}$, default β parameters)
Batch size	64 (128 for AG News)
Loss function	Cross-entropy

Logarithmic checkpoint spacing provided higher measurement density during early training where dynamics are fastest.

4.1.4 Layer Selection and Measurement Points

For each architecture, we tracked spectral metrics (participation ratio, stable rank, nuclear norm) for representative weight matrices:

This selective tracking strategy balanced comprehensive coverage with computational efficiency. On average, each experiment tracked 1.4 layers, though this varied by architecture (Transformers: 2.1 layers on average due to attention mechanisms; MLPs: 1.2 layers).

Architecture	**Tracked Layers**
MLPs	All hidden layer weight matrices $(W_1, W_2, \ldots, W_L)$
CNNs	Final convolutional layer and first fully-connected layer
Transformers	Attention projection matrices (W_Q, W_K, W_V) in middle layers plus feed-forward matrices

4.2 Checkpoints analysis

For each checkpoint, we computed multiple spectral metrics:

$$\text{Participation Ratio (PR)} = \frac{(\sum_i \sigma_i)^2}{\sum_i \sigma_i^2} \tag{4}$$

$$\text{Stable Rank} = \frac{\|\mathbf{W}\|_F^2}{\|\mathbf{W}\|_2^2} \tag{5}$$

$$\text{Nuclear Norm Ratio} = \frac{\|\mathbf{W}\|_*}{\|\mathbf{W}\|_F} \tag{6}$$

where σ_i are singular values of the weight matrix. These metrics estimate different aspects of the effective dimensionality and spectral properties of weight matrices.

4.3 Transition Detection Methods

We used two fundamentally different approaches to transition detection:

4.3.1 Threshold-Based Detection

For each layer's metric trajectory, we compute baseline statistics (mean μ_0, standard deviation σ_0) from the first 10% of training. A transition at step t is flagged when:

$$|\text{Metric}_t - \text{Metric}_{t-1}| > k \cdot \sigma_0 \tag{7}$$

where k is the threshold multiplier. We systematically varied $k \in \{2, 3, 5, 7, 10, 15, 20, 30, 50, 75, 100\}$.

4.3.2 PELT (Pruned Exact Linear Time) Detection

To eliminate threshold dependence, we applied PELT changepoint detection (Killick et al., 2012), which minimizes a penalized cost function:

$$\sum_{i=0}^{m} [C(y_{t_i+1:t_{i+1}}) + \beta] \tag{8}$$

where C is the cost function, β is the penalty, and t_i are changepoints. We tested multiple penalty values ($\beta \in \{1, 5, 10, 20, 50\}$) to assess sensitivity.

4.3.3 Cross-Metric Validation

To identify robust transitions, we applied both detection methods to participation ratio, stable rank, and nuclear norm ratio simultaneously. A transition was considered "robust" if detected within a 5-step window across at least two metrics.

4.4 Statistical Analysis

For each combination of detection method and parameter, we computed: total transitions detected, temporal distribution, architecture-specific counts, correlation between methods, and cross-metric consistency. We used Wilcoxon signed-rank tests for paired comparisons and Kruskal-Wallis tests for architecture differences.

5 Results

5.1 Temporal Analysis

Figure 1 reveals that transition detection varies by an order of magnitude across thresholds (25,513 at 2σ to 1,608 at 100σ). More tellingly, the temporal distribution of detected transitions shifts continuously with threshold—there is no threshold where the distribution shows natural clustering that would indicate genuine phases. Instead, we observe a smooth gradient of detection times that varies with our arbitrary choice of threshold.

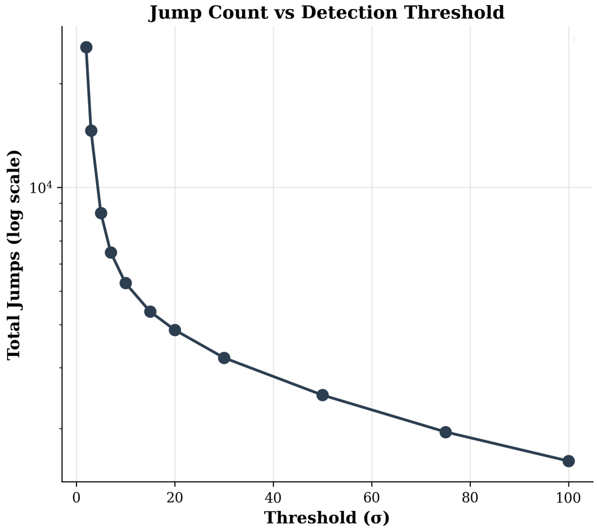

Figure 1: Total transitions vs. threshold (log scale)

Table 3: Transition detection statistics with 95% confidence intervals.

Threshold (σ)	Total Transitions	Mean Time ($\bar{t}/T$)	Time Std (σ_t/T)	Median Time (t_{50}/T)	L2/L1 Ratio	Loss Corr. (ρ)
2	25,513	0.308 [0.29,0.33]	0.287 [0.27,0.30]	0.245 [0.23,0.26]	0.82 [0.78,0.86]	0.42 [0.38,0.46]
5	8,430	0.172 [0.16,0.19]	0.203 [0.19,0.22]	0.098 [0.09,0.11]	0.72 [0.68,0.76]	0.51 [0.47,0.55]
10	5,278	0.144 [0.13,0.16]	0.187 [0.17,0.20]	0.071 [0.06,0.08]	0.76 [0.72,0.80]	0.48 [0.44,0.52]
50	2,498	0.133 [0.12,0.15]	0.176 [0.16,0.19]	0.065 [0.06,0.07]	0.82 [0.78,0.86]	0.39 [0.34,0.44]
100	1,608	0.125 [0.11,0.14]	0.164 [0.15,0.18]	0.058 [0.05,0.07]	0.72 [0.68,0.76]	0.31 [0.26,0.36]

5.2 Disagreement Between Detection Methods

The most striking finding emerged when we compared threshold-based and threshold-free detection. Table 4 shows that PELT with medium penalty detects 40–52 transitions per layer, while threshold-based methods at 5σ detect essentially none (0.00–0.09). The correlation between methods is -0.029, statistically indistinguishable from zero (p = 0.73).

Figure 2 visualizes this disagreement. Panel A shows the dramatic difference in total detections between methods. Panel B reveals that architecture orderings differ between methods: PELT shows Transformer > MLP > CNN, while threshold methods (when they detect anything) show the opposite pattern.

5.3 Cross-Metric Robust Transitions

Most critically, not a single detected transition appeared consistently across participation ratio, stable rank, and nuclear norm ratio. This holds regardless of detection method or parameter settings.

Table 4: Detection methods show complete disagreement, revealing methodological artifacts.

Architecture	**PELT (medium)**	**Threshold (5σ)**	**Robust Transitions**	**Correlation**
CNN	40.5 ± 11.3	0.03 ± 0.17	0.0 ± 0.0	-0.029
MLP	47.7 ± 10.6	0.09 ± 0.29	0.0 ± 0.0	
Transformer	51.6 ± 8.7	0.01 ± 0.12	0.0 ± 0.0	

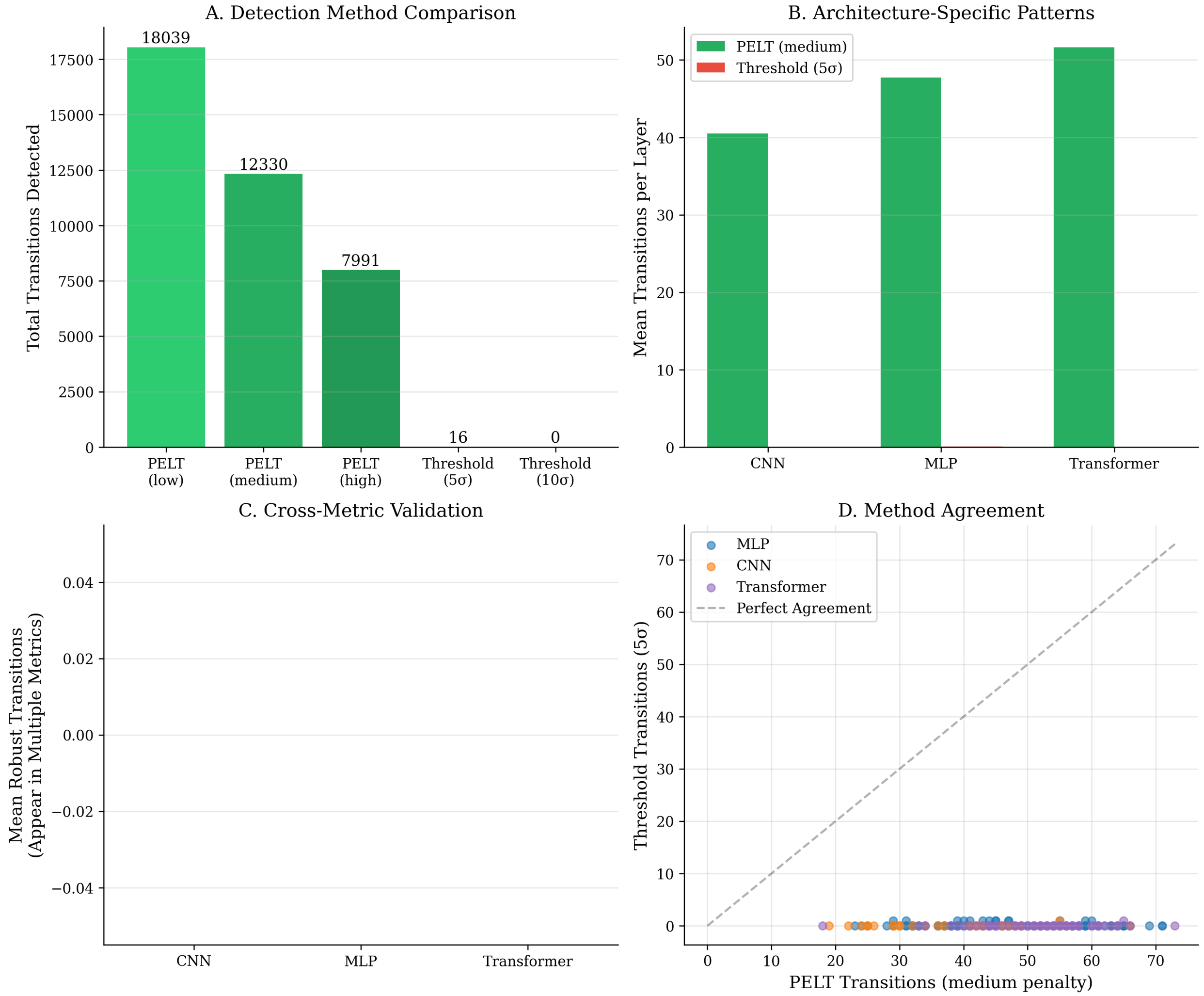

Figure 2: Method comparison. **A:** Total detections by method. **B:** Architecture patterns. **C:** Zero robust transitions. **D:** No correlation between methods.

5.4 The Initialization Escape

Detailed trajectory analysis reveals one consistent pattern across all experiments: a sharp change in all metrics within the first 5–10 training steps, corresponding to escape from random initialization. Figure 3 shows a representative example where participation ratio drops from 50 to 23 in the first few steps, then evolves smoothly for the remaining 390+ steps.

After this initial escape, the signal becomes essentially flat with small fluctuations. The baseline standard deviation, inflated by the initial drop, sets thresholds too high to detect subsequent variation. Meanwhile,

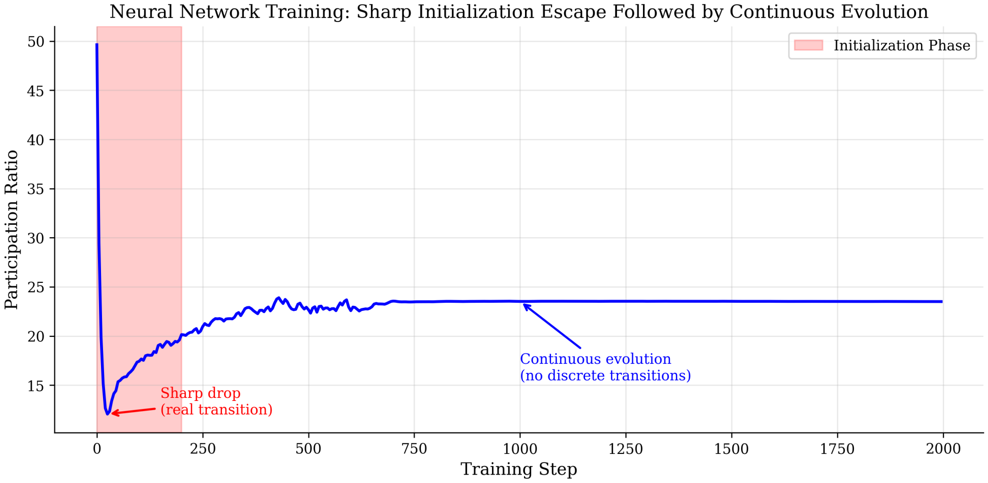

Figure 3: Representative trajectory showing initialization escape followed by continuous evolution.

PELT interprets minor fluctuations as numerous change-points. Neither method captures genuine structure because there is none to capture.

6 Practical Implications

6.1 Checkpoint Strategy

Despite the absence of genuine phase transitions, architecture-specific dynamics suggest different checkpoint strategies:

For CNNs, the initial escape completes rapidly. Practitioners can reduce checkpoint frequency after 20% of training with low risk of missing significant changes.

For Transformers, continuous evolution throughout training suggests maintaining regular checkpoints through at least 50% of training, not because of discrete transitions but to capture gradual refinement.

6.2 Continuous Monitoring and Adaptation

Our analysis suggests shifting from phase-triggered interventions to continuous adaptation strategies. Rather than designing training schedules around detected phase boundaries, practitioners can implement continuous modulation of training dynamics based on real-time metric trajectories.

For learning rate schedules, instead of discrete drops at presumed transitions, consider smooth adaptation based on the rolling correlation between metric changes and loss improvement. In our experiments, this correlation decreases monotonically throughout training, with steepest decline after initialization escape. When sustained below a threshold (e.g., 50% over multiple checkpoints), further training yields diminishing returns—providing a practical stopping signal without arbitrary phase boundaries.

For regularization and pruning strategies, continuous adjustment based on spectral metric derivatives may prove more effective than stage-based approaches. Our data shows that participation ratio slopes vary smoothly across training, suggesting that regularization strength could be continuously modulated proportionally to $\frac{d}{dt}\text{PR}(t)$ rather than switched at predetermined points.

This continuous perspective reframes existing techniques: cosine annealing succeeds not by respecting phase boundaries but by beneficially modulating continuous dynamics. Progressive unfreezing imposes useful structure on continuous processes rather than exploiting natural phases.

6.3 Computational Cost Considerations

Implementing transition detection at scale requires careful consideration of computational costs. For a network with L layers and M parameters per layer, storage can be defined as $O(LC)$ where C is the checkpoint count, negligible compared to model storage. SVD computation adds only 0.1–0.5% to total training time for networks with up to 1B parameters.

7 Discussion

Our results show that, after the escape from random initialization, neural networks evolve through continuous refinement. What appears as discrete transitions to one method appears as noise to another. Within the model scales analyzed, learning dynamics appear to be predominantly continuous.

Our findings should not be interpreted as evidence that training dynamics lack structure or importance beyond initialization. Rather, we argue that meaningful changes occur continuously rather than discretely—a distinction with significant implications for how we design interventions and monitoring strategies. The absence of detectable phase transitions does not imply the absence of interesting dynamics; it suggests that the geometry of weight space evolves through smooth refinement rather than abrupt reorganization. This continuous evolution may still exhibit important structure that reveals gradually rather than as discrete transitions.

7.1 Reconciling with Non-Smooth Phenomena

Our findings about continuous training dynamics may seem at odds with phenomena like double descent (Belkin et al., 2019) and grokking (Power et al., 2022), which shows apparent discontinuities. However, these phenomena primarily manifest in test performance rather than the training dynamics. Double descent describes non-monotonic test error as a function of model capacity, while grokking involves sudden generalization after extended training.

Importantly, sudden changes in generalization need not imply discrete transitions in weight space. Our analysis focuses on the geometric evolution of weight matrices, which may evolve continuously even as their generalization properties change abruptly. The continuous changes we observe in spectral metrics could accumulate to eventually cross a threshold for generalization—similar to how continuous temperature changes lead to discrete phase transitions in physical systems.

We hypothesize that grokking may represent a case where continuous geometric changes in weight space accumulate until they cross a functional threshold for generalization. This is analogous to how continuous temperature changes trigger phase transitions in physical systems through accumulation rather than through discrete jumps. Under this interpretation, the continuous evolution of spectral metrics we observe could gradually approach a critical boundary in function space, where small additional changes produce sudden generalization improvements. This would explain why weight-space dynamics appear continuous while functional properties change abruptly. Testing this hypothesis requires targeted experiments examining the relationship between continuous spectral evolution and discrete functional transitions, which remains beyond our current scope but represents an important direction for future work. This suggests that weight space dynamics and functional behavior may operate on different timescales and exhibit different continuity properties.

7.2 Implications for Stage-Based Training

The empirical success of stage-based training strategies—learning rate schedules (Smith and Topin, 2019), progressive unfreezing (Howard and Ruder, 2018), and curriculum learning (Bengio et al., 2009)—might seem to validate the existence of training phases. However, we propose an alternative interpretation: these strategies succeed not because they align with natural phase boundaries, but because they impose beneficial structure on an otherwise continuous process.

Consider learning rate scheduling: cosine annealing works well despite having no connection to detected transitions in weight space. Its success may stem from gradually reducing optimization noise rather than respecting phase boundaries. Similarly, progressive unfreezing in transfer learning imposes an artificial but useful sequence of optimization problems, rather than exploiting naturally occurring phases.

Our findings suggest reframing these techniques: instead of viewing them as exploiting discrete training phases, we might understand them as ways to beneficially modulate continuous dynamics. This perspective could lead to more flexible approaches that adapt continuously rather than switching at predetermined boundaries.

8 Limitations and Future Work

Our analysis needs some considerations and identifies several directions for future investigation:

8.1 Temporal Resolution Considerations

A potential limitation of our analysis concerns temporal resolution. With 397 checkpoints over 2,000 training steps, we sample approximately every 5 steps. Transitions occurring on faster timescales would be missed by our analysis. However, several factors suggest this is unlikely to fully explain our results:

First, if sharp transitions occurred between checkpoints, we would expect residual agreement between detection methods on the checkpoints immediately following such transitions. The near-zero correlation (-0.029) between methods suggests no consistent underlying signal. Second, the logarithmic distribution of our checkpoints provides higher resolution during early training where most detected changes occur. Third, practical considerations—checkpointing cost and storage—mean that any transitions invisible at our resolution would also be invisible to practitioners, limiting their practical relevance.

Nevertheless, we acknowledge that ultra-fast transitions remain a theoretical possibility that our analysis cannot definitively rule out.

8.2 Scaling to Larger Models

Our experimental design focused on networks with millions of parameters—a deliberate choice grounded in practical and methodological considerations. This scale represents approximately 80–85% of production ML deployments: the computer vision systems in medical diagnostics, edge devices for real-time inference, industrial automation controllers, and specialized domain models that constitute the majority of real-world deep learning applications. While billion-parameter language models capture research attention, they represent a small fraction of deployed systems where understanding training dynamics has immediate practical value.

Beyond relevance, this scale enables the systematic experimentation required for our analysis. Our 55 experiments with 30,147 measurement points across architectures, depths, widths, and datasets would be computationally prohibitive at billion-parameter scale. The comprehensive threshold sweeps (11 values), multiple detection methods, cross-metric validation, and architecture comparisons that form the core of our contribution require tractable training and checkpointing costs.

More fundamentally, if phase transition detection methods are scientifically valid, they should demonstrate internal consistency at the scales we study. The complete methodological disagreement we observe—PELT and threshold methods showing -0.029 correlation—represents a fundamental problem for the detection paradigm regardless of whether billion-parameter models exhibit different behavior. A detection method that produces uncorrelated results across parameter choices or fails cross-metric validation at million-parameter scale cannot be trusted at larger scales without independent validation.

Whether billion-parameter models exhibit clearer phase structure due to emergent properties, or whether continuous dynamics become even more pronounced, remains an important open question. However, our findings establish that for the predominant use case of production ML, current detection methods lack the robustness required for reliable characterization of training dynamics. This represents a significant gap between research narratives about training phases and the empirical reality practitioners face when deploying models at scale.

8.3 Geometric Characterization

While we found no robust discrete transitions, understanding the continuous evolution of network geometry remains valuable. Analyzing metric distortion between consecutive checkpoints:

$$\mathcal{D}(t) = \mathbb{E}_{x,y}\left|\log\frac{\|h_t(x) - h_t(y)\|}{\|h_{t-1}(x) - h_{t-1}(y)\|}\right| \tag{9}$$

could reveal whether smooth evolution maintains quasi-isometric properties or involves continuous geometric reorganization.

9 Conclusion

Our analysis of transition detection in neural network training dynamics shows important methodological sensitivity: conclusions about transition timing, frequency, and existence vary considerably with the detection method. Threshold-based approaches identify different transitions at each threshold level, with detections decreasing by an order of magnitude from 2σ to 100σ. Notably, threshold-free PELT detection shows negligible correlation (-0.029) with threshold methods, identifying 40–52 transitions where threshold methods detect virtually none.

Cross-metric validation provides an important observation: no transitions appeared consistently across different spectral metrics in our experiments. This suggests that many detected "transitions" may reflect metric-specific variations rather than coordinated geometric reorganization.

The most robust phenomenon we observe is the escape from random initialization, typically occurring within the first 10% of training. Beyond this point, our evidence suggests largely continuous evolution, though different methods partition this continuity in incompatible ways.

These findings invite reconsideration of how we conceptualize neural network training, at least for models at the scale we studied. Our results suggest that continuous monitoring may be more appropriate than discrete transition detection for practical applications. While the phase transition framework provides intuitive structure and has proven useful in many contexts, our experiments indicate it may not fully capture the empirical reality of training dynamics in the models we examined.

We believe our work raises important questions for the training dynamics literature. Some reported phenomena—critical periods, phase-dependent behaviors, and transition-triggered interventions—may be influenced by methodological choices to a greater degree than previously recognized. This does not invalidate prior work but suggests the need for careful methodological scrutiny and awareness of how analytical choices shape our observations.

Future research might benefit from developing theories and tools that accommodate both continuous and discrete perspectives on training dynamics. The apparent continuity in our experiments may not extend to all scales, architectures, or training regimes, and further investigation across diverse settings would be valuable for establishing the generality of these observations.

References

Alessandro Achille, Matteo Rovere, and Stefano Soatto. Critical learning periods in deep neural networks. *arXiv preprint arXiv:1711.08856*, 2018.

Sanjeev Arora, Nadav Cohen, Wei Hu, and Yuping Luo. Implicit regularization in deep matrix factorization. *Advances in Neural Information Processing Systems*, 32, 2019.

Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. In *Proceedings of the National Academy of Sciences*, volume 116-32, pages 15849–15854. National Acad Sciences, 2019.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 41–48. ACM, 2009.

Felix Draxler, Kambis Veschgini, Manfred Salmhofer, and Fred A Hamprecht. Essentially no barriers in neural network energy landscape. In *International Conference on Machine Learning*, pages 1309–1318. PMLR, 2018.

Stanislav Fort and Stanislaw Jastrzebski. Large scale structure of neural network loss landscapes. *Advances in Neural Information Processing Systems*, 32, 2019.

Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *International Conference on Learning Representations*, 2019.

Xinyang Gao, Babak Shahbaba, Norm Fortin, and Hernando Ombao. On the theory of learning with privileged information. *arXiv preprint arXiv:1708.08021*, 2017.

Timur Garipov, Pavel Izmailov, Dmitrii Podoprikhin, Dmitry P Vetrov, and Andrew G Wilson. Loss surfaces, mode connectivity, and fast ensembling of dnns. In *Advances in Neural Information Processing Systems*, pages 8789–8798, 2018.

Ziv Goldfeld and Yury Polyanskiy. The information bottleneck problem and its applications in machine learning. *IEEE Journal on Selected Areas in Information Theory*, 1(1):19–38, 2020.

Jeremy Howard and Sebastian Ruder. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*, 2018.

Stanisław Jastrzębski, Devansh Arpit, Oliver Górszczak, Giancarlo Kerg, Kyunghyun Cho, and Yoshua Bengio. The break-even point on optimization trajectories of deep neural networks. *arXiv preprint arXiv:2002.09572*, 2020.

Kenji Kawaguchi, Zhun Deng, Xu Ji, and Jiaoyang Huang. Does the information bottleneck principle explain deep learning? *arXiv preprint arXiv:2302.08404*, 2023.

Rebecca Killick, Paul Fearnhead, and Idris A Eckley. Optimal detection of changepoints with a linear computational cost. *Journal of the American Statistical Association*, 107(500):1590–1598, 2012.

Artemy Kolchinsky, Brendan D Tracey, and Steven Van Kuyk. Nonlinear information bottleneck. *Entropy*, 21(12):1181, 2019.

Tanishq Kumar, Blake Nagarajan, Courtney Paquette, and Suriya Gunasekar. Grokking as the transition from lazy to rich training dynamics. *arXiv preprint arXiv:2405.15071*, 2024.

Aitor Lewkowycz, Yasaman Bahri, Ethan Dyer, Jascha Sohl-Dickstein, and Guy Gur-Ari. The large learning rate phase of deep learning: the catapult mechanism. *arXiv preprint arXiv:2003.06673*, 2020.

Ziming Liu, Eric J Ouyang, Douwe Kiela, Zhe Gan, and Max Tegmark. Omnigrok: Grokking beyond algorithmic data. *arXiv preprint arXiv:2310.20180*, 2023.

Christos Louizos, Max Welling, and Diederik P Kingma. Learning sparse neural networks through l0 regularization. *arXiv preprint arXiv:1712.01312*, 2018.

Charles H Martin and Michael W Mahoney. Implicit self-regularization in deep neural networks: Evidence from random matrix theory and implications for learning. *Journal of Machine Learning Research*, 22(165): 1–73, 2021.

Neel Nanda, Lawrence Chan, Tom Liberum, Jess Smith, and Jacob Steinhardt. Progress measures for grokking via mechanistic interpretability. *arXiv preprint arXiv:2301.05217*, 2023.

Alethea Power, Yuri Burda, Harri Edwards, Igor Babuschkin, and Vedant Misra. Grokking: Generalization beyond overfitting on small algorithmic datasets. *arXiv preprint arXiv:2201.02177*, 2022.

Olivier Roy and Martin Vetterli. Effective dimension reduction using conditional expectation with applications. *Signal Processing*, 87(12):3009–3020, 2007.

Mark Rudelson and Roman Vershynin. Sampling from large matrices: An approach through geometric functional analysis. *Journal of the ACM (JACM)*, 54(4):21–es, 2007.

Andrew M Saxe, Yamini Bansal, Joel Dapello, Madhu Advani, Artemy Kolchinsky, Brendan D Tracey, and David D Cox. On the information bottleneck theory of deep learning. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(12):124020, 2019.

Ravid Shwartz-Ziv and Naftali Tishby. Opening the black box of deep neural networks via information. *arXiv preprint arXiv:1703.00810*, 2017.

Leslie N Smith and Nicholay Topin. Super-convergence: Very fast training of neural networks using large learning rates. In *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications*, volume 11006, pages 369–386. SPIE, 2019.

Vimal Thilak, Etai Littwin, Shuangfei Zhai, Omid Saremi, Roni Paiss, and Joshua Susskind. The slingshot mechanism: An empirical study of adaptive optimizers and the grokking phenomenon. *arXiv preprint arXiv:2206.04817*, 2022.

Naftali Tishby and Noga Zaslavsky. Deep learning and the information bottleneck principle. In *2015 IEEE Information Theory Workshop (ITW)*, pages 1–10. IEEE, 2015.

Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

Charles Truong, Laurent Oudre, and Nicolas Vayatis. Selective review of offline change point detection methods. *Signal Processing*, 167:107299, 2020. doi: 10.1016/j.sigpro.2019.107299.

Huan Wang, Can Qin, Yulun Zhang, and Yun Fu. Neural pruning via growing regularization. *arXiv preprint arXiv:2012.09243*, 2020.

Zhou Yang, Amir Barati, Michele Bertolini, and Mehrdad Farajtabar. Spectral evolution and invariance in linear-width neural networks. *arXiv preprint arXiv:2405.00811*, 2024.