ON THE RESIDUAL SCALING OF LOOPED TRANSFORMERS: STABILITY AND TRANSFERABILITY

Shaowen Wang, Bingrui Li, Ge Zhang, Wenhao Huang, Jian Li[†]
wangsw23@mails.tsinghua.edu.cn, lijian83@mail.tsinghua.edu.cn

ABSTRACT

Looped (weight-tied) Transformers increase effective depth by repeatedly applying a shared block for L steps. In practice, larger L often improves capability, but requires careful hyperparameter tuning. We study the parameterization of pre-norm looped Transformers and ask which residual scaling enables stable training and transferable hyperparameters across loop counts. In contrast to the common $1/\sqrt{L}$ scale in deep networks, our simplified tied-weight residual MLP analysis shows that looped models require $1/L$ residual scaling. We validate theoratical predictions on a standard pre-norm Transformer architecture. Our experiments with looped LLMs across various loop times and learning rates demonstrate that $1/L$ scaling offers significantly better stability and hyperparameter transfer than $1/\sqrt{L}$ scaling

1 INTRODUCTION

Looped (weight-tied) Transformers offer a compelling parameter-efficient scaling paradigm: a single shared block is applied for L steps, increasing effective depth without inflating the parameter count. This design has appeared in Universal Transformers (Dehghani et al., 2018), ALBERT (Lan et al., 2019), and recent work on algorithmic reasoning and latent computation (Yang et al., 2023; Fan et al., 2024; Saunshi et al., 2025; Gatmiry et al., 2024; Ng & Wang, 2024; Zhu et al., 2025). By decoupling depth from memory footprint, looped architectures provide a pathway to multi-step reasoning within a compact parameter space.

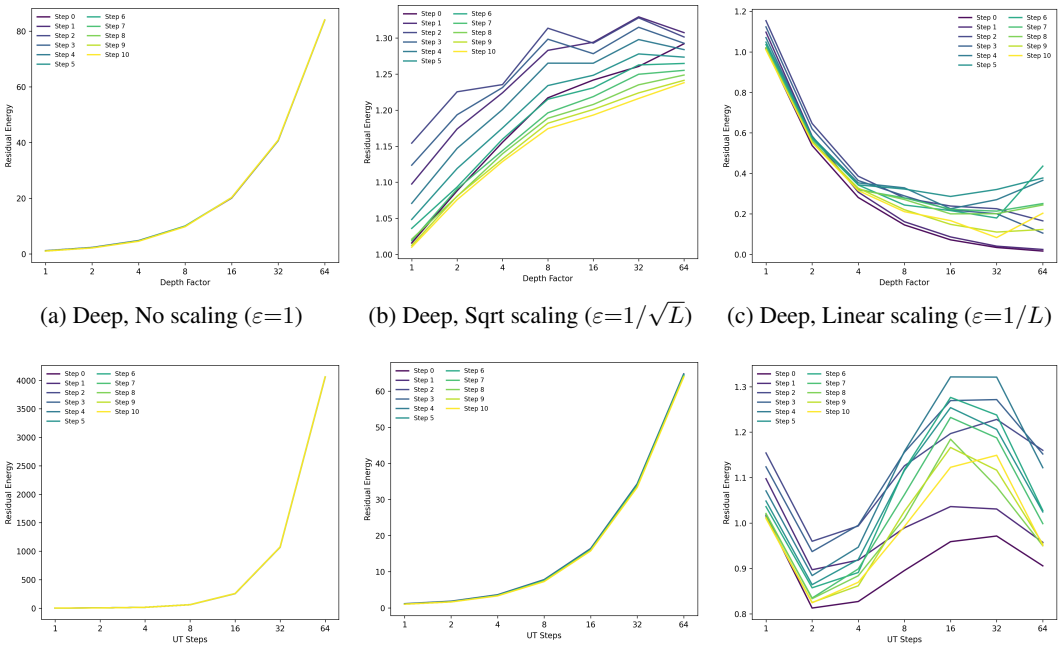
However, optimization becomes difficult as the loop count L increases, leading to training instability, exploding gradients, and high sensitivity to the learning rate (Zhu et al., 2025). Consequently, finding the optimal learning rate for large L is necessary but computationally expensive. This raises a critical question: Can we tune hyperparameters at small loop steps and transfer them to large L to achieve stable training without costly retuning?

To answer this, we first examine whether scaling rules designed for standard deep networks can naturally adapt to looped architectures (Yang et al., 2022; Hayou & Yang, 2023; Dey et al., 2025; Mlodozeniec et al., 2025). Since a looped model is functionally equivalent to a deep network with shared weights, one might expect standard depth-scaling heuristics to apply. In non-shared residual networks, scaling residual branches by $1/\sqrt{L}$ controls hidden variance because independent weights across layers lead to random-walk (linear) variance accumulation (Bordelon et al., 2023).

However, our analysis of looped MLPs shows that this structural independence assumption collapses. Weight sharing induces strong correlations between residual updates at different steps, causing the variance magnitude to grow quadratically with L rather than linearly. Consequently, the standard $1/\sqrt{L}$ scaling is insufficient for looped architectures. Based on this insight, we derive the necessary stability condition. To counteract the quadratic accumulation, looped models require *linear residual scaling*—specifically, $\varepsilon = L^{-\alpha}$ with $\alpha \geq 1$. This is stricter than the $\alpha \geq 1/2$ threshold sufficient for non-shared deep networks (Yang et al., 2022; Dey et al., 2025).

Following the one-step update analysis in Dey et al. (2025), we derive that the optimal learning rate scales as $\eta \propto L^{\alpha-1}$, consistent with deep model behavior. Crucially, under our proposed linear scaling ($\alpha = 1$), this simplifies to a constant learning rate ($\eta \propto L^0$). This property allows for seamless

[†]Corresponding author.



(a) Deep, No scaling (ε=1) (b) Deep, Sqrt scaling (ε=$1/\sqrt{L}$) (c) Deep, Linear scaling (ε=$1/L$)

(d) Looped, No scaling (ε=1) (e) Looped, Sqrt scaling (ε=$1/\sqrt{L}$) (f) Looped, Linear scaling (ε=$1/L$)

Figure 1: **Initialization stability test across different architectures and scaling rules.** We measure the residual energy (residual stream variance) H (vertical axis) against the depth factor or loop count L (horizontal axis) during the first 10 optimization steps. The key observation is that while standard $\sqrt{L}$ scaling stabilizes deep networks (b), it fails to control variance growth in looped models (e). In contrast, linear scaling effectively maintains energy invariance across increasing loop counts in looped architectures (f).

transfer: a model tuned at $L = 1$ can be directly scaled to larger L without any modification to the learning rate, dramatically reducing experimental costs.

Our contribution is as follows:

- Variance analysis: We show that, unlike standard deep networks, looped models exhibit quadratic variance growth due to weight sharing. We demonstrate that linear residual scaling ($\alpha \geq 1$) is required to counteract this growth and stabilize the forward pass.
- Learning Rate Scaling rule: We derive the learning rate scaling $\eta \propto L^{1-\alpha}$ for loop transformers. For the proposed linear scaling case ($\alpha = 1$), this implies a constant learning rate, enabling effective hyperparameter transfer from shallow to deep loop settings.
- Empirical Validation: We empirically verify these findings on looped Transformers trained with various loop counts ($L = 1$ to 8) on Fineweb-edu (Penedo et al., 2024) and initialization stability test up to $L = 64$. Our method yields consistent training stability and predictable hyperparameter behavior compared to conventional scaling heuristics.

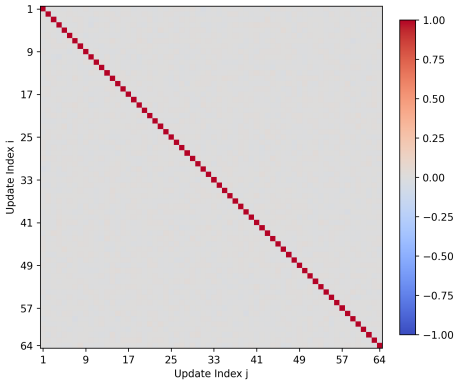
2 Why loops require linear residual scaling

We study a tied-weight residual model that captures the key loop effect:

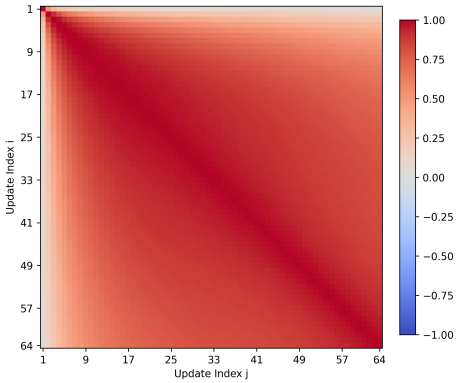
$$h_{\ell+1} = h_\ell + \varepsilon W \phi(h_\ell), \qquad \varepsilon = L^{-\alpha},\ \ell = 0, \ldots, L-1, \tag{1}$$

with $h_\ell \in \mathbb{R}^N$, shared $W \in \mathbb{R}^{N \times N}$, ReLU ϕ, and initialization $W_{ij} \sim \mathcal{N}(0, \sigma_W^2/N)$. Define the residual energy of the l-th layer H_l as the variance of the residual stream:

$$H_\ell \triangleq \frac{1}{N} \|h_\ell\|_2^2, \qquad r_\ell \triangleq W\phi(h_\ell).$$



(a) Deep (Non-shared): Correlations are concentrated near the diagonal.



(b) Looped (Shared): Correlations remain positive over off-diagonal pairs.

Figure 2: **Pairwise correlation matrices for residual increments (r_i, r_j) across update indices.** These heatmaps visualize the cross-term contributions to the total variance, defined as $C_L = \frac{\varepsilon^2}{N} \sum_{i,j} \langle r_i, r_j \rangle$. While deep networks (a) show negligible off-diagonal interaction, looped models (b) exhibit a dense correlation pattern where a large fraction of (i, j) terms contribute. This persistent correlation is the empirical mechanism driving the quadratic $\Theta(\varepsilon^2 L^2)$ variance accumulation in weight-tied architectures.

2.1 Residual-energy scaling in loops

Unrolling (1),

$$h_L = h_0 + \varepsilon \sum_{\ell=0}^{L-1} r_\ell.$$

So

$$H_L = H_0 + \underbrace{\frac{2\varepsilon}{N} \sum_{\ell=0}^{L-1} \langle h_0, r_\ell \rangle}_{B_L} + \underbrace{\frac{\varepsilon^2}{N} \sum_{\ell=0}^{L-1} \sum_{s=0}^{L-1} \langle r_\ell, r_s \rangle}_{C_L}. \tag{2}$$

This decomposition isolates the correlation term C_L, whose behavior is visualized in Figure 2. The heatmaps empirically confirm that while off-diagonal terms $\langle r_\ell, r_s \rangle$ vanish in deep networks, they remain significant in looped architectures due to weight sharing. We will formally derive the mathematical mechanism behind this non-cancellation in the subsequent analysis.

Theorem 1 (Looped residual-energy scaling law)**.** *Under the standard mean-field approximations:*

1. *wide-limit Gaussian statistics with $H_\ell = O(1)$,*
2. *mean-field reduction $W^\top W \approx \sigma_W^2 I$,*
3. *positive average cross-step ReLU overlap $\frac{1}{L^2} \sum_{\ell,s} \mathbb{E}[C_{\ell,s}] = \Theta(1)$ where $C_{\ell,s} \triangleq \frac{1}{N} \langle \phi(h_\ell), \phi(h_s) \rangle$,*

we have

$$\mathbb{E}[H_L] = H_0 + O(\varepsilon L) + \Theta(\varepsilon^2 L^2).$$

Therefore, to keep residual energy bounded as $L \to \infty$, one needs

$$\varepsilon L = O(1), \qquad \textit{i.e.} \qquad \alpha \geq 1.$$

Proof. See Appendix C.

Table 1: Residual scaling and learning-rate transfer along depth/loop axes.

Model family	Weight sharing?	Stable energy condition	LR transfer rule
Deep residual stack	No	$\alpha \geq \frac{1}{2}$ (App. B)	$\eta \propto L^{\alpha-1}$ (Dey et al., 2025)
Looped residual stack	Yes	$\alpha \geq 1$ (Thm. 1)	$\eta \propto L^{\alpha-1}(2)$

2.2 Learning-rate scaling for one-step updates

Our goal is to determine the scaling rule for the learning rate η that ensures stable training dynamics across different loop counts L. Specifically, we require that for a given small weight update ΔW, the resulting change in the model's final output, Δh_L, remains order $\Theta(1)$ (independent of L). If Δh_L grows with L, updates will explode in deep networks; if it vanishes, training will stall.

For one optimizer step $t \to t+1$, define

$$\Delta W(t) = W(t+1) - W(t), \qquad \Delta h_\ell(t) = h_\ell(t+1) - h_\ell(t).$$

Theorem 2 (Loop-wise learning-rate scaling)**.** *Assume:*

1. *sign-normalized update scale* $\Delta W_{ij} = \frac{\eta}{\sqrt{N}} S_{ij}$ *and* $\|\Delta W\| = \Theta(\eta)$,
2. *stable activations* $\|\phi(h_\ell)\|_2^2 / N = \Theta(1)$,
3. *stable forward scaling* $\varepsilon L = O(1)$.

Then

$$\frac{1}{N}\mathbb{E}[\|\Delta h_L\|_2^2] = \Theta\big((\eta\varepsilon L)^2\big) \quad \textit{(up to L-independent constants)}. \tag{3}$$

So depth-invariant one-step output change requires

$$\eta\,\varepsilon\,L = \Theta(1) \quad \Longleftrightarrow \quad \eta \propto L^{\alpha-1}. \tag{4}$$

Proof. See Appendix D.

At the critical loop scaling $\alpha = 1$, (4) gives $\eta = \Theta(1)$, matching our transfer behavior in experiments.

3 Experiments

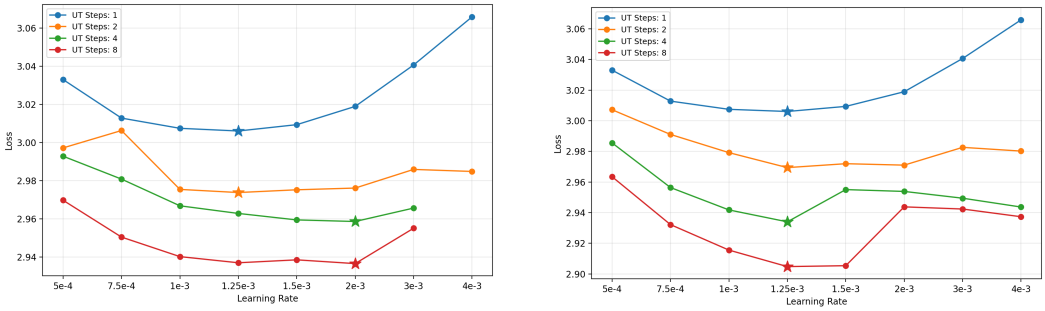
We evaluate two claims from Theorems 1 and 2: linear residual scaling stabilizes looped Transformers, and it improves hyperparameter transfer across loop counts. All models are decoder-only and use a Llama-style pre-norm Transformer block (RMSNorm), with looping applied to all Transformer blocks except the token embedding and the unembedding head.

3.1 Residual-energy diagnostic (small-step training)

Protocol. We run a controlled experiment that measures residual energy during the first 10 optimizer steps. We vary the number of loop iterations $L \in \{1, 2, 4, 8, 16, 32, 64\}$ and compare three residual scalings: *none* (ε=1), *sqrt* (ε=$1/\sqrt{L}$), and *linear* (ε=$1/L$). Training uses Adam for 10 steps, batch size 1, sequence length 128, and fixed random token inputs. We record residual energy H at the end of the last loop iteration (right before the final RMSNorm), and report averages over 10 seeds.

Findings. Consistent with Theorem 1, $1/\sqrt{L}$ scaling is insufficient for looped models: residual energy grows rapidly with L and can explode during early optimization. In contrast, linear scaling keeps the residual energy bounded and approximately invariant across L. For non-shared deep stacks, $1/\sqrt{L}$ scaling stabilizes residual energy, while $1/L$ tends to over-dampen it.

Figure 2 directly matches the mechanism in Theorem 1: in deep non-shared stacks, the effective contribution pattern is close to diagonal; in looped shared stacks, many off-diagonal pairs are non-negligible. Therefore, in loops, C_L is large because many pairwise terms $\langle r_\ell, r_s \rangle$ contribute, not only the diagonal terms.



(a) Sqrt scaling ($\varepsilon = 1/\sqrt{L}$)

(b) Linear scaling ($\varepsilon = 1/L$)

Figure 3: **Learning-rate transfer comparison for 340M looped language models.** Each curve represents a different loop count $L \in \{1, 2, 4, 8\}$, with star markers indicating the optimal learning rate for that L. Runs resulting in training divergence are omitted for clarity. Under standard sqrt scaling (a), the optimal learning rate shifts significantly as L increases. In contrast, under our proposed linear scaling (b), the optima remain aligned in a narrow band across all loop counts, demonstrating robust hyperparameter transfer.

3.2 Hyperparameter transfer in language-model training (340M)

Setup. We train a 340M-parameter decoder-only language model with loop counts $L \in \{1, 2, 4, 8\}$. Training data is FineWeb-Edu with a total training budget of 10B tokens: 20,000 optimizer steps with a global batch size of 0.5M tokens. We use Adam with $(\beta_1, \beta_2) = (0.9, 0.95)$ and decoupled weight decay chosen so that lr $\times$ wd $= 10^{-5}$. The learning-rate schedule is warmup-stable-linear-decay, with 500 warmup steps and a final 1,000-step linear decay. For each L, we sweep a small grid of learning rates around a base value tuned at L=1. We compare sqrt vs. linear residual scaling.

Results. We observe two key advantages of linear residual scaling. (i) Improved Performance: At $L = 8$, linear scaling achieves a significantly lower minimum loss than sqrt scaling (a non-trivial reduction of 0.025), indicating better trainability at larger depths. (ii) Predictable Transfer: As shown in Figure 3, the optimal learning rate for linear scaling remains nearly invariant across $L \in \{1, 2, 4, 8\}$, consistent with our theoretical prediction ($\eta \propto L^0$). In contrast, the optimal learning rate for sqrt scaling shifts significantly as L increases and deviates from standard deep-network scaling rules (Dey et al., 2025), complicating hyperparameter selection.

4 Conclusion

Looped Transformers offer a promising path toward parameter-efficient reasoning, yet their adoption has been hindered by severe optimization instabilities at larger depths. Our work identifies the root cause of this bottleneck: unlike standard deep networks, weight sharing induces strong cross-step correlations that drive a quadratic explosion in residual energy. We demonstrate that this structural difference necessitates a stricter linear residual scaling ($\varepsilon \propto 1/L$), rather than the standard root-scaling used in deep learning. Beyond stability, this scaling rule unlocks a predictable hyperparameter transfer regime ($\alpha = 1$), allowing learning rates tuned at $L = 1$ to transfer seamlessly to deeper loop configurations. By resolving these fundamental optimization issues, our findings pave the way for more robust and scalable looped architectures in large-scale language modeling.

Future work may extend these findings by investigating the training dynamics of looped models beyond initialization, as well as their behavior across alternative architectures such as Post-Norm Transformers. Examining how shared representations evolve during training will be essential for further refining these scaling laws.

Acknowledgments

We thank Kaifeng Lv, Kaiyue Wen, and Tansheng Zhu for helpful discussions.

REFERENCES

Thomas Bachlechner, Bodhisattwa Prasad Majumder, Huanru Henry Mao, Garrison W. Cottrell, and Julian McAuley. Rezero is all you need: Fast convergence at large depth. 2020. URL https://arxiv.org/abs/2003.04887.

Blake Bordelon, Lorenzo Noci, Mufan Bill Li, Boris Hanin, and Cengiz Pehlevan. Depthwise hyperparameter transfer in residual networks: Dynamics and scaling limit. 2023. doi: 10.48550/ARXIV.2309.16620. URL https://arxiv.org/abs/2309.16620.

Mostafa Dehghani, Stephan Gouws, O. Vinyals, Jakob Uszkoreit, and Lukasz Kaiser. Universal transformers. *ArXiv*, abs/1807.03819, 2018.

Nolan Dey, Bin Claire Zhang, Lorenzo Noci, Mufan Li, Blake Bordelon, Shane Bergsma, Cengiz Pehlevan, Boris Hanin, and Joel Hestness. Don't be lazy: Completep enables compute-efficient deep transformers. 2025. doi: 10.48550/ARXIV.2505.01618. URL https://arxiv.org/abs/2505.01618.

Ying Fan, Yilun Du, Kannan Ramchandran, and Kangwook Lee. Looped transformers for length generalization. 2024. doi: 10.48550/ARXIV.2409.15647. URL https://arxiv.org/abs/2409.15647.

Khashayar Gatmiry, Nikunj Saunshi, Sashank J. Reddi, Stefanie Jegelka, and Sanjiv Kumar. Can looped transformers learn to implement multi-step gradient descent for in-context learning? 2024. doi: 10.48550/ARXIV.2410.08292. URL https://arxiv.org/abs/2410.08292.

Soufiane Hayou and Greg Yang. Width and depth limits commute in residual networks. 2023. doi: 10.48550/ARXIV.2302.00453. URL https://arxiv.org/abs/2302.00453.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations. *ArXiv*, abs/1909.11942, 2019.

Pierre Marion. Generalization bounds for neural ordinary differential equations and deep residual networks. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/98ed250b203d1ac6b24bbcf263e3d4a7-Abstract-Conference.html.

Pierre Marion, Adeline Fermanian, Gérard Biau, and Jean-Philippe Vert. Scaling resnets in the large-depth regime. 2025. URL https://arxiv.org/abs/2206.06929.

Bruno Mlodozeniec, Pierre Ablin, Louis Béthune, Dan Busbridge, Michal Klein, Jason Ramapuram, and Marco Cuturi. Completed hyperparameter transfer across modules, width, depth, batch and duration. 2025. URL https://arxiv.org/abs/2512.22382.

Kei-Sing Ng and Qingchen Wang. Loop neural networks for parameter sharing. 2024. URL https://arxiv.org/abs/2409.14199.

Guilherme Penedo, Hynek Kydlíček, Anton Lozhkov, Margaret Mitchell, Colin A Raffel, Leandro Von Werra, Thomas Wolf, et al. The fineweb datasets: Decanting the web for the finest text data at scale. *Advances in Neural Information Processing Systems*, 37:30811–30849, 2024.

Nikunj Saunshi, Nishanth Dikkala, Zhiyuan Li, Sanjiv Kumar, and Sashank J. Reddi. Reasoning with latent thoughts: On the power of looped transformers. 2025. URL https://arxiv.org/abs/2502.17416.

Hongyu Wang, Shuming Ma, Li Dong, Shaohan Huang, Dongdong Zhang, and Furu Wei. Deepnet: Scaling transformers to 1,000 layers. 2022. URL https://arxiv.org/abs/2203.00555.

Greg Yang, Edward J. Hu, Igor Babuschkin, Szymon Sidor, Xiaodong Liu, David Farhi, Nick Ryder, Jakub Pachocki, Weizhu Chen, and Jianfeng Gao. Tensor programs v: Tuning large neural networks via zero-shot hyperparameter transfer. 2022. doi: 10.48550/ARXIV.2203.03466. URL https://arxiv.org/abs/2203.03466.

Liu Yang, Kangwook Lee, Robert Nowak, and Dimitris Papailiopoulos. Looped transformers are better at learning learning algorithms. 2023. doi: 10.48550/ARXIV.2311.12424. URL https://arxiv.org/abs/2311.12424.

Hongyi Zhang, Yann Dauphin, and Tengyu Ma. Fixup initialization: Residual learning without normalization. *ArXiv*, abs/1901.09321, 2019.

Rui-Jie Zhu, Zixuan Wang, Kai Hua, Tianyu Zhang, Ziniu Li, Haoran Que, Boyi Wei, Zixin Wen, Fan Yin, He Xing, Lu Li, Jiajun Shi, Kaijing Ma, Shanda Li, Taylor Kergan, Andrew Smith, Xingwei Qu, Mude Hui, Bohong Wu, Qiyang Min, Hongzhi Huang, Xun Zhou, Wei Ye, Jiaheng Liu, Jian Yang, Yunfeng Shi, Chenghua Lin, Enduo Zhao, Tianle Cai, Ge Zhang, Wenhao Huang, Yoshua Bengio, and Jason Eshraghian. Scaling latent reasoning via looped language models. 2025. doi: 10.48550/ARXIV.2510.25741. URL https://arxiv.org/abs/2510.25741.

A Related Work

Looped and parameter-shared Transformers. Using one block repeatedly has been explored as recurrent depth in Universal Transformers (Dehghani et al., 2018) and as cross-layer parameter sharing in ALBERT (Lan et al., 2019). More recent looped models show strong behavior on iterative algorithm learning and multi-step in-context procedures (Yang et al., 2023; Gatmiry et al., 2024), length generalization (Fan et al., 2024), and latent-reasoning style test-time compute scaling (Saunshi et al., 2025; Zhu et al., 2025). Related parameter-sharing formulations also appear in loop neural networks (Ng & Wang, 2024). Our work is aligned with this line, but focuses on a specific missing piece: a scaling law for stable optimization and hyperparameter transfer as loop count changes.

Stability in deep residual stacks and deep Transformers. Large-depth training has a long line of stabilization techniques, including residual reparameterizations and initialization rules such as Fixup and ReZero (Zhang et al., 2019; Bachlechner et al., 2020), and Transformer-specific stabilizers such as DeepNet/DeepNorm (Wang et al., 2022). On the theory side, prior analyses of deep non-shared residual networks characterize when residual scaling keeps signals controlled in the large-depth limit (Marion et al., 2025). A complementary line studies generalization for continuous-depth models and their ResNet analogues: Marion (2023) derives a Lipschitz-based bound whose complexity term depends on differences between successive weight matrices. These works mostly treat depth as a stack of distinct layers. Our setting is different: loop steps reuse the same parameters, which changes the interaction structure across steps.

Hyperparameter transfer and parameterization. Tensor-program and μP-style analyses establish transfer principles across model scales and motivate systematic parameterization choices (Yang et al., 2022). Recent depth-transfer analyses in residual networks study how learning-rate and initialization choices change with depth under non-shared assumptions (Bordelon et al., 2023; Hayou & Yang, 2023). CompleteP and follow-up work extend this direction for deep Transformers and broader axes of transfer (Dey et al., 2025; Mlodozeniec et al., 2025). We view our result as complementary: we target the *loop axis* specifically and show that shared weights create strong cross-step correlations, leading to a different stability threshold and transfer regime from standard depth scaling.

B Deep (non-shared) residual stacks

For comparison, consider

$$h_{\ell+1} = h_\ell + \varepsilon W_\ell \phi(h_\ell), \qquad W_\ell \text{ independent across } \ell.$$

Let $H_\ell = \|h_\ell\|_2^2 / N$. Expanding one step gives

$$H_{\ell+1} = H_\ell + \frac{2\varepsilon}{N} \langle h_\ell, W_\ell \phi(h_\ell) \rangle + \frac{\varepsilon^2}{N} \|W_\ell \phi(h_\ell)\|_2^2 .$$

Because h_ℓ depends on $W_{<\ell}$ but not on W_ℓ, the cross term has zero mean. Under standard mean-field scaling, the last term is $\Theta(H_\ell)$, so

$$\mathbb{E}[H_{\ell+1}] \approx (1 + c\varepsilon^2)\mathbb{E}[H_\ell], \qquad c = \Theta(1).$$

Iterating this recursion:

$$\mathbb{E}[H_L] \approx (1 + c\varepsilon^2)^L H_0 \approx \exp(cL\varepsilon^2) H_0.$$

With $\varepsilon = L^{-\alpha}$:

$$\mathbb{E}[H_L] \approx \exp(cL^{1-2\alpha}) H_0.$$

Hence bounded energy for large L requires $\alpha \geq \frac{1}{2}$.

C LOOPED RESIDUAL-ENERGY PROOF

For the looped model

$$h_{\ell+1} = h_\ell + \varepsilon W \phi(h_\ell), \qquad r_\ell \triangleq W\phi(h_\ell), \qquad H_\ell \triangleq \frac{1}{N} \|h_\ell\|_2^2,$$

define

$$B_L \triangleq \frac{2\varepsilon}{N} \sum_{\ell=0}^{L-1} \langle h_0, r_\ell \rangle, \qquad C_L \triangleq \frac{\varepsilon^2}{N} \sum_{\ell=0}^{L-1} \sum_{s=0}^{L-1} \langle r_\ell, r_s \rangle.$$

Unrolling the recursion gives $h_L = h_0 + \varepsilon \sum_{\ell=0}^{L-1} r_\ell$, hence

$$H_L = H_0 + B_L + C_L. \tag{5}$$

C.1 WHY THE QUADRATIC TERM APPEARS

Write $u_\ell \triangleq \phi(h_\ell)$. Then

$$\frac{1}{N} \langle r_\ell, r_s \rangle = \frac{1}{N} u_\ell^\top W^\top W u_s.$$

Under the same large-width approximation used in the main text, $W^\top W \approx \sigma_W^2 I$, so

$$\frac{1}{N} \langle r_\ell, r_s \rangle \approx \sigma_W^2 \frac{1}{N} \langle u_\ell, u_s \rangle = \sigma_W^2 C_{\ell,s},$$

where

$$C_{\ell,s} \triangleq \frac{1}{N} \langle \phi(h_\ell), \phi(h_s) \rangle.$$

Therefore

$$C_L \approx \sigma_W^2 \varepsilon^2 \sum_{\ell,s=0}^{L-1} C_{\ell,s}.$$

If the average overlap is order-one positive,

$$\frac{1}{L^2} \sum_{\ell,s=0}^{L-1} \mathbb{E}[C_{\ell,s}] = \Theta(1),$$

then immediately

$$\mathbb{E}[C_L] = \Theta(\varepsilon^2 L^2).$$

For ReLU-Gaussian coordinates, this positivity condition is natural:

$$\mathbb{E}[\mathrm{ReLU}(Z_1)\mathrm{ReLU}(Z_2)] = \sqrt{H_1 H_2}\, \kappa(\rho),$$

with

$$\kappa(\rho) = \frac{1}{2\pi} \left(\sqrt{1-\rho^2} + (\pi - \arccos\rho)\rho \right) \in [0, 1/2].$$

Except the degenerate case $\rho = -1$, , $\kappa(\rho) > 0$.

C.2 WHY THE LINEAR TERM IS SMALLER

For each ℓ, Cauchy gives

$$\frac{1}{N}\big|\,\langle h_0, r_\ell\rangle\,\big| \le \sqrt{H_0}\sqrt{\frac{1}{N}\,\|r_\ell\|_2^2}.$$

So

$$|B_L| \le 2\varepsilon\sqrt{H_0}\sum_{\ell=0}^{L-1}\sqrt{\frac{1}{N}\,\|r_\ell\|_2^2}.$$

Under standard initialization,

$$\mathbb{E}\left[\frac{1}{N}\,\|r_\ell\|_2^2\right] \approx \sigma_W^2\,\mathbb{E}[\mathrm{ReLU}(Z_\ell)^2] = \sigma_W^2\,\frac{H_\ell}{2} = O(1),$$

as long as $H_\ell = O(1)$. Hence $|B_L| = O(\varepsilon L)$.

C.3 PROOF OF THEOREM 1

Proof. Combine (5) with the two bounds above:

$$\mathbb{E}[H_L] = H_0 + O(\varepsilon L) + \Theta(\varepsilon^2 L^2).$$

For bounded residual energy as $L \to \infty$, the dominant term must satisfy $\varepsilon^2 L^2 = O(1)$, equivalently $\varepsilon L = O(1)$. With $\varepsilon = L^{-\alpha}$, this gives $\alpha \ge 1$. □

D LEARNING-RATE SCALING PROOF

Fix one update $t \to t+1$:

$$\Delta W \triangleq W(t+1) - W(t), \qquad \Delta h_\ell \triangleq h_\ell(t+1) - h_\ell(t).$$

Write $h_\ell^+ = h_\ell(t+1)$ and $h_\ell = h_\ell(t)$. Subtracting the two forward recursions gives

$$\Delta h_{\ell+1} = \Delta h_\ell + \varepsilon\Delta W\phi(h_\ell^+) + \varepsilon W\big(\phi(h_\ell^+) - \phi(h_\ell)\big). \tag{6}$$

Summing over ℓ yields

$$\Delta h_L = \varepsilon\Delta W V + \varepsilon W\Delta U + R_2, \tag{7}$$

where

$$V = \sum_{\ell=0}^{L-1}\phi(h_\ell), \quad \Delta U = \sum_{\ell=0}^{L-1}\big(\phi(h_\ell^+) - \phi(h_\ell)\big), \quad R_2 = \varepsilon\Delta W\sum_{\ell=0}^{L-1}\big(\phi(h_\ell^+) - \phi(h_\ell)\big).$$

We omit R_2 because it's a second order term.

D.1 SCALE OF THE LEADING TERM

Let $u_\ell = \phi(h_\ell)$. For $V = \sum_{\ell=0}^{L-1} u_\ell$, Cauchy gives

$$\frac{1}{N}\,\|V\|_2^2 \le L\sum_{\ell=0}^{L-1}\frac{1}{N}\,\|u_\ell\|_2^2 = O(L^2),$$

using $\mathbb{E}[\|u_\ell\|_2^2/N] = \mathbb{E}[\mathrm{ReLU}(Z_\ell)^2] = O(1)$. For the lower bound, $\mathbb{E}[\mathrm{ReLU}(Z_\ell)] > 0$ implies a linear growth along the all-ones direction, so

$$\frac{1}{N}\,\|V\|_2^2 = \Omega(L^2).$$

Hence

$$\frac{1}{N}\,\|V\|_2^2 = \Theta(L^2), \qquad \|V\|_2 = \Theta(\sqrt{N}\,L).$$

Now let $T_{\text{main}} = \varepsilon \Delta W V$ and use $\Delta W_{ij} = \frac{\eta}{\sqrt{N}} S_{ij}$ with $\|\Delta W\| = \Theta(\eta)$. Then

$$\|T_{\text{main}}\|_2 \leq \varepsilon \|\Delta W\| \|V\|_2 = O(\eta \varepsilon L \sqrt{N}),$$

which gives

$$\frac{1}{N} \mathbb{E}[\|T_{\text{main}}\|_2^2] = O(\eta^2 \varepsilon^2 L^2).$$

Under isotropic high-dimensional updates, this bound is tight in scaling, so

$$\frac{1}{N} \mathbb{E}[\|T_{\text{main}}\|_2^2] = \Theta\big((\eta \varepsilon L)^2\big).$$

D.2 Nonlinear correction does not change the exponent

Define $a_\ell = \|\Delta h_\ell\|_2$, $c = \|W\|$, $d = \|\Delta W\|$. Because ReLU is 1-Lipschitz,

$$\big\|\phi(h_\ell^+) - \phi(h_\ell)\big\|_2 \leq a_\ell.$$

Applying triangle inequality and operator norms to (6):

$$a_{\ell+1} \leq (1 + \varepsilon c) a_\ell + \varepsilon d \big\|\phi(h_\ell^+)\big\|_2.$$

With $\Delta h_0 = 0$, unrolling gives

$$a_L \leq \varepsilon \sum_{\ell=0}^{L-1} (1 + \varepsilon c)^{L-1-\ell} d \big\|\phi(h_\ell^+)\big\|_2.$$

If $\varepsilon L = O(1)$, $c = O(1)$, $d = \Theta(\eta)$, and $\big\|\phi(h_\ell^+)\big\|_2 = O(\sqrt{N})$, then

$$\|\Delta h_L\|_2 = O(\eta \varepsilon L \sqrt{N}),$$

and therefore

$$\frac{1}{N} \|\varepsilon W \Delta U\|_2^2 = O(\eta^2 \varepsilon^2 L^2).$$

So this correction has the same L exponent as the leading term. The remainder R_2 in (7) is second order in small updates and does not change the scaling exponent.

D.3 Proof of Theorem 2

Proof. From (7), the leading term scales as $\Theta((\eta \varepsilon L)^2)$ in normalized squared norm, while the nonlinear correction has the same or smaller depth exponent. Hence

$$\frac{1}{N} \mathbb{E}[\|\Delta h_L\|_2^2] = \Theta\big((\eta \varepsilon L)^2\big) \quad \text{(up to } L\text{-independent constants)}.$$

Keeping one-step output perturbation invariant across L requires $\eta \varepsilon L = \Theta(1)$, i.e. $\eta \propto L^{\alpha - 1}$. □