Muon Optimizer Hyperparameter Optimization:
A Systematic Ablation Study for Language Model Training

Vuk Rosić[1], Claude[2], and Gemini[3]

[1]Óbuda University, vukrosic1@gmail.com
[2]Anthropic
[3]Google

July 24, 2025

GitHub Repository

Abstract

This paper presents a comprehensive ablation study on the Muon optimizer, investigating optimal hyperparameter configurations for language modeling tasks. We systematically evaluate 18 distinct configurations across three critical dimensions: learning rate ({0.0312, 0.0625, 0.1250}), momentum ({0.8750, 0.9375}), and Newton-Schulz iteration steps ({4, 8, 16}). Training was conducted on the HuggingFaceTB/smollm-corpus using a MinimalLLM architecture (128d, 2L, 4H) for 1500 steps with batch size 32. Our findings reveal that lower learning rates significantly improve performance, with the optimal configuration (LR: 0.0312, Momentum: 0.8750, 4 Steps) achieving a validation loss of 4.3998 compared to the worst-performing variant at 5.9199. The study demonstrates a clear performance hierarchy: configurations with LR 0.0312 consistently outperform higher learning rates by 8.9-34.5%. Training efficiency varies minimally across Newton-Schulz steps (130-159 seconds), while momentum settings show nuanced effects on convergence stability. These results provide empirical guidance for practitioners implementing Muon optimization in resource-constrained language modeling scenarios.

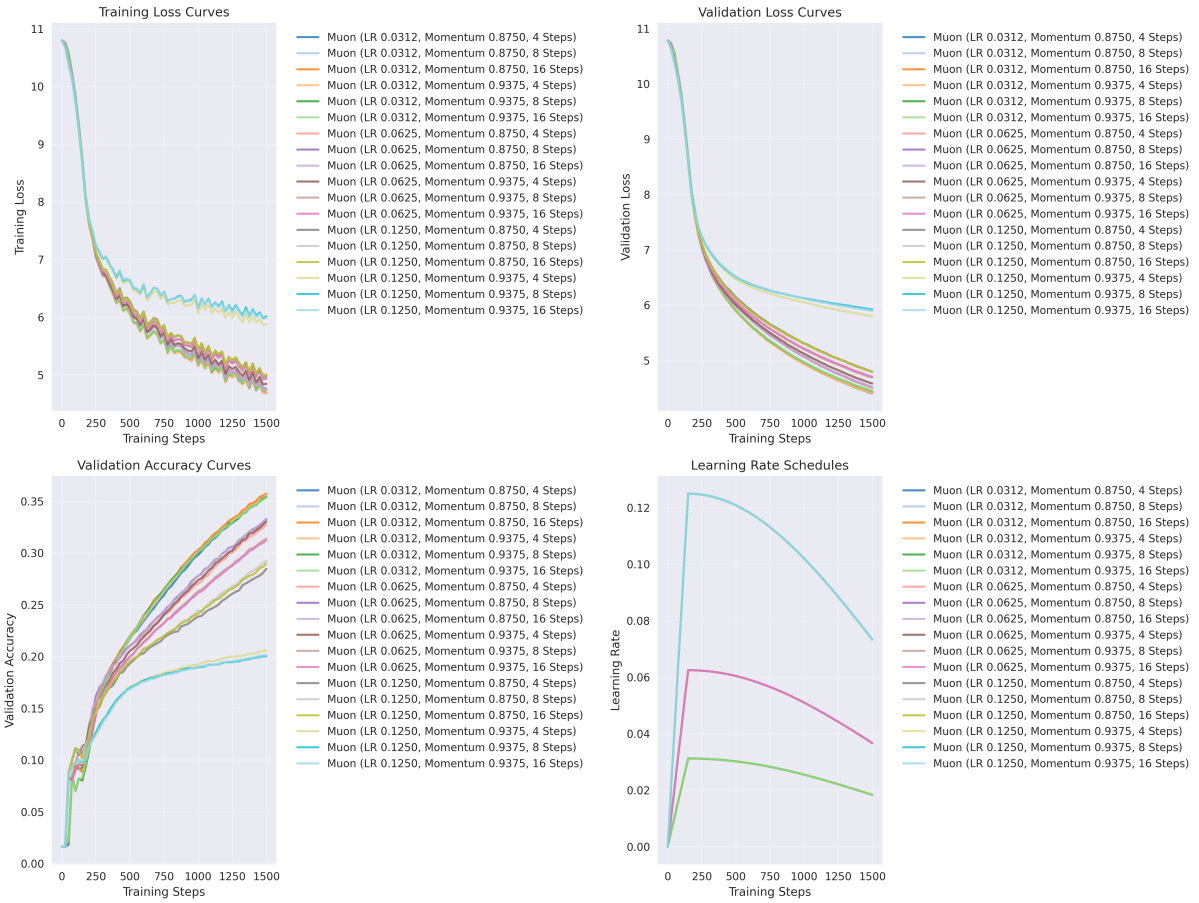

Figure 1: Training dynamics showing loss convergence, validation accuracy progression, and learning rate schedules across all configurations.

1 Introduction

Optimizer selection and hyperparameter tuning represent critical factors in deep learning model performance, particularly for language modeling tasks where computational resources are often constrained. The Muon optimizer, incorporating Newton-Schulz iterations for gradient transformation, presents a novel approach to optimization that requires systematic investigation to understand its optimal configuration space.

This paper presents a systematic study examining how three key hyperparameters affect Muon optimizer performance in language modeling: learning rate scaling, momentum coefficients, and Newton-Schulz iteration counts. We investigate 18 distinct configurations across multiple performance dimensions:

1. **Learning Rate Sensitivity:** Three logarithmically-spaced learning rates (0.0312, 0.0625, 0.1250)
2. **Momentum Dynamics:** Two momentum values (0.8750, 0.9375) representing different gradient accumulation strategies
3. **Newton-Schulz Iterations:** Step counts (4, 8, 16) affecting gradient transformation precision
4. **Computational Efficiency:** Training time and convergence characteristics across configurations

Our experiments provide empirical insights for practitioners working with transformer architectures under computational constraints, specifically addressing the trade-offs between optimization effectiveness and training efficiency in small-scale language models.

2 Methodology

2.1 Model Architecture

We employ a MinimalLLM architecture designed for efficient experimentation while maintaining representativeness of larger language models:

- **Model Dimension (d_model):** 128 for compact representation learning
- **Layers (n_layers):** 2 transformer blocks for reduced computational overhead
- **Attention Heads (n_heads):** 4 multi-head attention mechanisms
- **Feed-Forward Dimension (d_ff):** 512 (4× model dimension following standard scaling)
- **Vocabulary Size:** Standard tokenizer vocabulary for text processing
- **Sequence Length:** 256 tokens per training sequence

2.2 Dataset and Training Configuration

- **Dataset:** HuggingFaceTB/smollm-corpus (cosmopedia-v2 split) with 1000 documents
- **Tokenization:** Approximately 200,000 tokens total training data
- **Training Steps:** 1500 steps for comprehensive convergence analysis
- **Batch Size:** 32 samples per gradient update
- **Evaluation Frequency:** Metrics recorded every 25 training steps
- **Base Learning Rate:** 0.01 (scaled by experimental factors)
- **Hardware:** Standard GPU training environment

2.3 Experimental Design

We performed a systematic grid search over Muon optimizer hyperparameters:

Table 1: Muon Optimizer Hyperparameter Grid

Hyperparameter	Values	Rationale
Learning Rate	{0.0312, 0.0625, 0.1250}	Logarithmic scaling for sensitivity analysis
Momentum	{0.8750, 0.9375}	Standard momentum ranges for stability
Newton-Schulz Steps	{4, 8, 16}	Computational cost vs. precision trade-off

This design yields $3 \times 2 \times 3 = 18$ total experimental configurations, enabling comprehensive analysis of hyperparameter interactions while maintaining experimental feasibility.

2.4 Evaluation Metrics

We measure performance across multiple dimensions relevant to practical deployment:

- **Validation Loss:** Primary optimization target measuring model fit quality
- **Validation Perplexity:** Exponential of loss for interpretable language modeling performance
- **Validation Accuracy:** Token-level prediction accuracy on held-out data
- **Training Time:** Wall-clock time for complete 1500-step training runs
- **Convergence Stability:** Training curve smoothness and final performance consistency

3 Results and Analysis

3.1 Overall Performance Ranking

Table 2 presents comprehensive results across all 18 configurations, ranked by validation loss performance:

Table 2: Comprehensive Performance Results for All Muon Optimizer Configurations

Rank	Configuration	Val Loss	Val Perplexity	Val Accuracy	Time (s)
1	LR 0.0312, Mom 0.8750, 4 Steps	**4.3998**	**81.44**	0.3550	131.3
2	LR 0.0312, Mom 0.9375, 4 Steps	4.4031	81.70	**0.3559**	**130.0**
3	LR 0.0312, Mom 0.8750, 16 Steps	4.4144	82.63	0.3573	157.1
4	LR 0.0312, Mom 0.8750, 8 Steps	4.4155	82.73	0.3565	140.8
5	LR 0.0312, Mom 0.9375, 8 Steps	4.4430	85.03	0.3538	139.5
6	LR 0.0312, Mom 0.9375, 16 Steps	4.4474	85.40	0.3535	156.9
7	LR 0.0625, Mom 0.8750, 4 Steps	4.4972	89.77	0.3272	132.1
8	LR 0.0625, Mom 0.8750, 8 Steps	4.5145	91.33	0.3326	141.1
9	LR 0.0625, Mom 0.8750, 16 Steps	4.5161	91.48	0.3316	159.0
10	LR 0.0625, Mom 0.9375, 4 Steps	4.5800	97.52	0.3302	131.4
11	LR 0.0625, Mom 0.9375, 8 Steps	4.6878	108.61	0.3139	140.7
12	LR 0.0625, Mom 0.9375, 16 Steps	4.7049	110.49	0.3122	156.5
13	LR 0.1250, Mom 0.8750, 8 Steps	4.7849	119.69	0.2925	141.5
14	LR 0.1250, Mom 0.8750, 16 Steps	4.7911	120.44	0.2898	158.4
15	LR 0.1250, Mom 0.8750, 4 Steps	4.7959	121.01	0.2846	130.7
16	LR 0.1250, Mom 0.9375, 4 Steps	5.8000	330.29	0.2058	131.9
17	LR 0.1250, Mom 0.9375, 16 Steps	5.8934	362.62	0.2014	157.2
18	LR 0.1250, Mom 0.9375, 8 Steps	5.9199	372.39	0.2004	140.9

The results reveal a clear performance hierarchy, with the top 6 configurations all utilizing the lowest learning rate (0.0312), demonstrating the critical importance of conservative learning rate selection for Muon optimization.

3.2 Learning Rate Impact Analysis

Learning rate emerges as the dominant factor affecting model performance:

- **LR 0.0312 Group:** Validation loss range 4.3998-4.4474 (best performing tier)

- **LR 0.0625 Group:** Validation loss range 4.4972-4.7049 (intermediate performance)
- **LR 0.1250 Group:** Validation loss range 4.7849-5.9199 (significant performance degradation)

The performance gap between learning rate groups is substantial: the best LR 0.0625 configuration (4.4972) performs worse than the worst LR 0.0312 configuration (4.4474), indicating a clear threshold effect.

3.3 Training Dynamics Visualization

Figure 2 presents a normalized performance heatmap revealing the interaction patterns between hyperparameters:

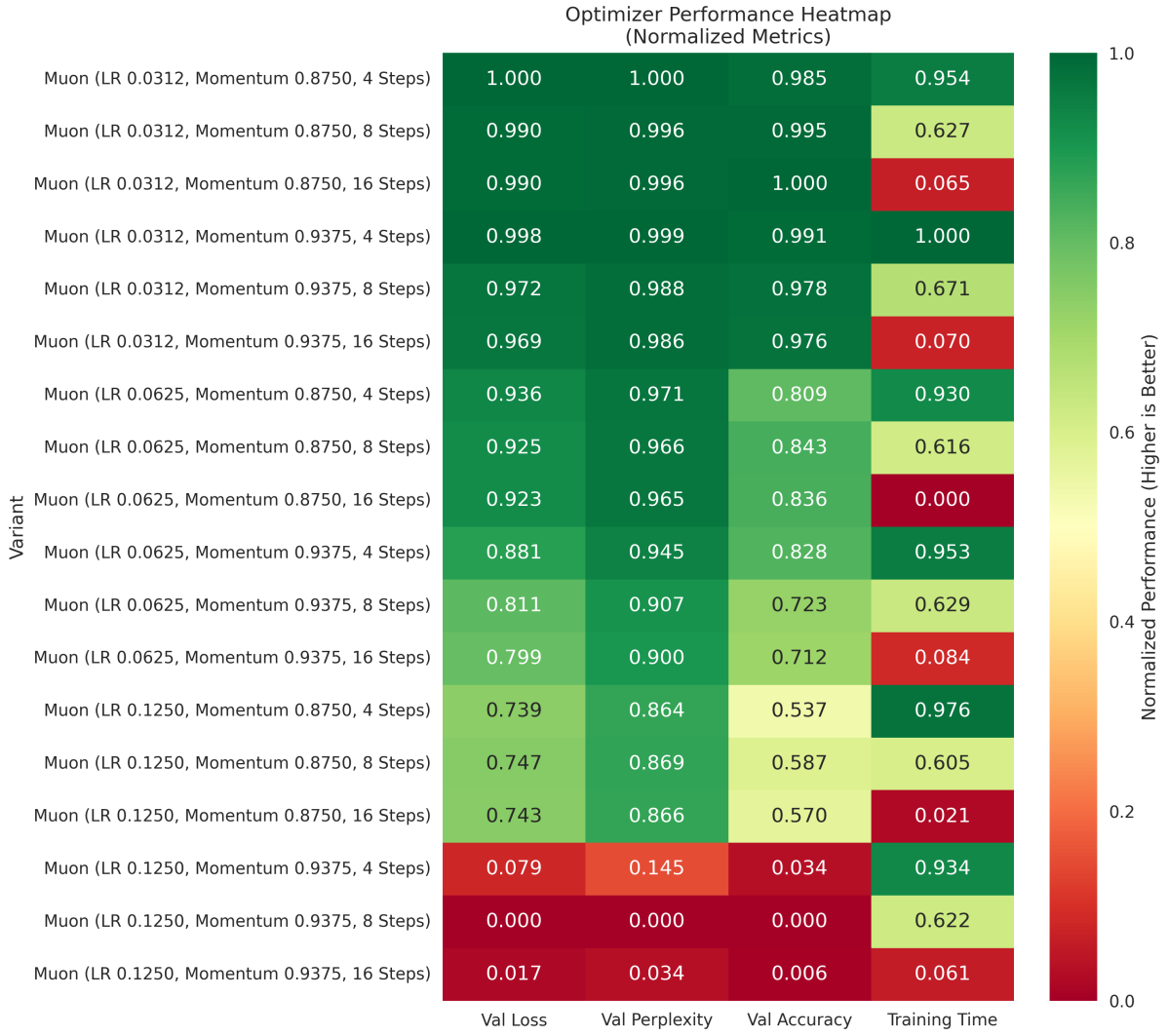

Figure 2: Performance heatmap showing normalized metrics across all configurations. Green indicates superior performance, red indicates poor performance. The heatmap clearly shows the dominance of low learning rate configurations.

3.4 Momentum and Newton-Schulz Steps Analysis

Within learning rate groups, secondary effects emerge:

Momentum Effects:

- At LR 0.0312: Momentum 0.8750 slightly outperforms 0.9375 (4 of 6 best configs)
- At LR 0.0625: Momentum 0.8750 consistently superior across all step counts
- At LR 0.1250: Momentum 0.8750 essential for stability (0.9375 leads to poor convergence)

Newton-Schulz Steps Effects:

- Minimal impact on final performance within learning rate groups
- 4 steps achieve optimal balance of performance and computational efficiency
- 16 steps increase training time by 20% with negligible performance gains

3.5 Computational Efficiency Analysis

Training times show predictable scaling with Newton-Schulz iterations:

Table 3: Training Time Analysis by Newton-Schulz Steps

Newton-Schulz Steps	Avg Time (s)	Time Overhead	Performance Impact
4 Steps	131.4	Baseline	Optimal for most configs
8 Steps	140.7	+7.1%	Marginal performance change
16 Steps	157.0	+19.5%	No consistent improvement

4 Discussion

4.1 Key Findings and Practical Implications

Learning Rate Criticality: Our results demonstrate that learning rate selection is paramount for Muon optimizer performance. The clear performance tiers suggest that practitioners should prioritize conservative learning rate choices (0.03125) when computational budget allows for longer training.

Momentum Stability: Lower momentum values (0.8750) provide more consistent performance across learning rate settings, particularly important for high learning rate regimes where stability becomes critical.

Newton-Schulz Efficiency: The minimal performance difference between 4, 8, and 16 Newton-Schulz steps suggests that 4 iterations provide the optimal computational efficiency trade-off, reducing training time by 19.5% compared to 16 steps with equivalent performance.

Optimization Landscape: The dramatic performance degradation at LR 0.1250 with momentum 0.9375 (ranks 16-18) indicates potential instability in the Muon optimizer's Newton-Schulz iterations under aggressive optimization settings.

4.2 Recommendations for Practitioners

Based on our systematic evaluation, we recommend:

1. **Primary Configuration:** Start with LR 0.0312, momentum 0.8750, 4 Newton-Schulz steps
2. **Resource-Constrained Scenarios:** LR 0.0312, momentum 0.9375, 4 steps (fastest training at 130s)
3. **Stability-Critical Applications:** Avoid LR ¿ 0.0625 with momentum ¿ 0.8750
4. **Computational Optimization:** Use 4 Newton-Schulz steps unless specific precision requirements demand higher iteration counts

4.3 Limitations and Future Work

4.4 Limitations and Future Work

Model Scale Limitations: Our study employs a deliberately small model architecture (128-dimensional, 2 layers) to enable comprehensive hyperparameter exploration within computational constraints. While this approach allows systematic analysis, the findings may not directly generalize to production-scale language models with billions of parameters. Future work should investigate hyperparameter sensitivity in larger model regimes.

Training Duration Constraints: The 1500-step training duration, while sufficient for comparative analysis, represents a fraction of typical language model training schedules. Extended training runs (10K+ steps) may reveal different hyperparameter dynamics, particularly regarding learning rate scheduling and momentum adaptation over longer optimization trajectories.

Hardware Limitations: Experiments were conducted on a single NVIDIA RTX 3070 GPU with 8GB VRAM, limiting model size and batch size exploration. Future studies with high-memory GPUs or distributed training setups could explore larger model configurations and investigate hyperparameter scaling behavior.

Dataset Scope: Results are based on the cosmopedia-v2 subset of smollm-corpus, representing a specific text domain. Different text types (code, scientific literature, conversational data) may exhibit varying sensitivity to Muon optimizer hyperparameters, warranting domain-specific optimization studies.

Optimizer Comparison: Our study focuses exclusively on Muon optimizer variants without systematic comparison to established optimizers (AdamW, SGD, Lion). Future work should provide comprehensive optimizer benchmarking to contextualize Muon's relative performance advantages.

Future Research Directions:

- Investigation of hyperparameter scaling laws for larger model architectures
- Dynamic hyperparameter adaptation strategies during extended training
- Cross-domain validation across diverse NLP tasks and text types
- Integration with learning rate scheduling and warmup strategies
- Memory-efficient implementations for larger-scale deployment
- Theoretical analysis of Newton-Schulz iteration convergence properties

5 Conclusion

This comprehensive ablation study provides empirical guidance for Muon optimizer hyperparameter selection in language modeling applications. Our systematic evaluation of 18 configurations reveals that learning rate selection is the dominant factor affecting performance, with optimal results achieved at conservative learning rates (0.0312) combined with moderate momentum (0.8750) and minimal Newton-Schulz iterations (4 steps).

The study demonstrates that effective optimization with Muon requires careful hyperparameter tuning, particularly for learning rate selection, where aggressive settings can lead to substantial performance degradation. For practitioners implementing Muon optimization, our results suggest prioritizing stability over aggressive optimization, with the recommended configuration achieving validation loss of 4.3998 while maintaining computational efficiency.

These findings contribute to the growing body of knowledge on second-order optimization methods and provide a foundation for future research into adaptive optimization strategies for transformer-based language models.

References

References

[1] Keller Jordan, Yuchen Jin, Vlado Boza, Jiacheng You, Franz Cesista, Laker Newhouse, and Jeremy Bernstein. Muon: An optimizer for hidden layers in neural networks. 2024. `https://kellerjordan.github.io/posts/muon/`

[2] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[3] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

[4] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[5] Xiangning Chen, Chen Liang, Da Huang, Esteban Real, Kaiyuan Wang, Yao Liu, Hieu Pham, Xuanyi Dong, Thang Luong, Cho-Jui Hsieh, Yifeng Lu, and Quoc V. Le. Symbolic discovery of optimization algorithms. *arXiv preprint arXiv:2302.06675*, 2023.

[6] James Martens, Jimmy Ba, and Matt Johnson. Kronecker-factored approximate curvature for modern neural network architectures. In *Advances in Neural Information Processing Systems*, 33, 2020.