Analysis of the Muon Optimizer in a Transformer Mixture-of-Experts Language Model

Vuk Rosić[1,2], [To Be Determined]
[1]Open Superintelligence Lab
[2]Óbuda University

September 12, 2025

Abstract

This paper documents a series of experiments analyzing the Muon optimizer for training a Mixture-of-Experts (MoE) Large Language Model. As part of an ongoing research effort, we conduct three primary experiments: a direct baseline comparison against the AdamW optimizer, an ablation study of Muon's core components (Momentum and Newton-Schulz orthogonalization), and a hyperparameter sensitivity analysis. We present the preliminary results from these experiments, which characterize the optimizer's behavior under various conditions. This work is part of an active research project, and final conclusions are yet to be drawn. The aim is to provide a transparent account of our methodology and current findings.

GitHub Repository **Research Discord**
This research is actively developed and discussed on the linked Discord server.

1 Introduction

The field of Large Language Models (LLMs) has been dominated by optimizers from the Adam family, particularly AdamW [1, 3]. While effective, these methods can face challenges in stability and efficiency when scaling to increasingly large models. The Muon optimizer has emerged as a promising alternative, leveraging matrix-aware techniques, specifically Newton-Schulz orthogonalization, to potentially offer a more stable and efficient training process [2, 5, 6].

This paper investigates the practical performance of the Muon optimizer on a Mixture-of-Experts (MoE) transformer-based LLM. We aim to answer the following research questions:

- How does a hybrid Muon optimizer compare to the standard AdamW optimizer in terms of performance and computational cost?
- What are the individual contributions of Muon's core components—momentum and Newton-Schulz orthogonalization?
- How sensitive is the Muon optimizer to its key hyperparameters?

To address these questions, we conduct a series of controlled experiments on a consistent model architecture and dataset.

2 Background

2.1 Mixture-of-Experts (MoE) Models

MoE models utilize a sparse activation strategy where only a fraction of the model's parameters are used for any given input [4]. This is achieved through a "router" network that directs input tokens to a subset of "expert" networks. This architecture allows for a significant increase in model capacity without a proportional increase in computational cost per forward pass, making it an efficient choice for large-scale models.

2.2 The Muon Optimizer

The Muon optimizer, short for MomentUm Orthogonalized by Newton-schulz, operates on entire matrices of parameters. Its key innovation is the use of the Newton-Schulz iteration to efficiently compute the orthogonalization of the gradient matrix [7]. This process helps to maintain gradient stability and encourages a more uniform learning process across the network's layers. In our experiments, we employ a hybrid strategy where Muon is applied to the 2D weight matrices of the model, while AdamW is used for other parameters like embeddings and normalization layers.

The core update step involves:

1. Applying momentum to the gradient.
2. Orthogonalizing the resulting gradient using the `zeropower_via_newtonschulz5` function.
3. Applying the final update to the parameter weights.

3 Experimental Setup

All experiments were conducted using a consistent setup to ensure comparability of results.

- **Model Architecture**: A 6-layer MoE Transformer with a model dimension of 384, 8 attention heads, and a feed-forward dimension of 1536. The model included 8 experts, with a top-2 routing strategy.
- **Dataset**: A 500,000 token subset of the SmolLM corpus.
- **Training**: All models were trained for 1000 steps with a batch size of 24 and 4 gradient accumulation steps.
- **Hardware**: Experiments were conducted in a low-compute environment, utilizing a single NVIDIA RTX 4090 GPU or a Google Colab T4 GPU. Future experiments may be upgraded to higher-compute environments.

4 Results and Analysis

4.1 Experiment 1: Baseline Comparison (Muon vs. AdamW)

This experiment compared the performance of the hybrid Muon optimizer against a pure AdamW optimizer.

The results in Table 1 show that the Muon optimizer achieved a slightly better validation loss, accuracy, and perplexity. However, this came at the cost of a 12.7% increase in training time compared to AdamW.

Table 1: Experiment 1: Muon vs. AdamW Performance.

Metric	Muon	AdamW	Difference
Validation Loss	**0.0476**	0.0547	-0.0072
Validation Accuracy	**0.9907**	0.9881	+0.0026
Validation Perplexity	**1.05**	1.06	-0.01
Training Time (min)	13.3	**11.8**	+1.5

4.2 Experiment 2: Ablation Study

This experiment analyzed the contribution of Muon's two main components: momentum and Newton-Schulz (NS) orthogonalization.

Table 2: Experiment 2: Ablation Study Results.

Variant	Val Loss	Val Acc	Val PPL	Time (min)
Full Muon (Momentum + NS)	**2.5347**	**0.4948**	**12.61**	2.7
Momentum Only (No NS)	5.4336	0.1385	228.98	**2.4**
NS Only (No Momentum)	3.8273	0.2926	45.94	2.7
Basic SGD-like (No Both)	5.2608	0.1628	192.63	**2.4**

The ablation study (Table 2) clearly demonstrates that both components are crucial. Removing either one led to a significant degradation in performance. The analysis indicated a strong, positive synergy effect of 1.4654, suggesting the components are more effective together than the sum of their individual contributions. Momentum appears to be the more critical component, as its removal resulted in a larger performance drop than the removal of Newton-Schulz.

4.3 Experiment 3: Hyperparameter Sensitivity

This experiment tested Muon's sensitivity to learning rate, momentum, and the number of Newton-Schulz steps.

Table 3: Experiment 3: Optimal Hyperparameters.

Hyperparameter	Optimal Value (based on Val Loss)
Learning Rate	0.05 (Loss: 0.3277)
Momentum	0.95 (Loss: 2.5296)
Newton-Schulz Steps	7 (Loss: 2.4955)

The optimizer showed the highest sensitivity to the **learning rate**, with performance improving by 18.6x from the worst to the best-tested value. A higher learning rate of 0.05 was found to be optimal. The model was moderately sensitive to **momentum**, with 0.95 being the optimal value and 0.99 leading to degraded performance. Sensitivity to the number of **Newton-Schulz steps** was the weakest, with 7 steps performing best, but showing diminishing returns compared to 5 steps.

5 Summary of Results and Future Work

The experiments conducted provide preliminary data on the Muon optimizer's behavior. This research is ongoing, and the results presented here form the basis for further investigation.

5.1 Summary of Current Results

- **Baseline Comparison**: The hybrid Muon optimizer achieved a marginally lower validation loss (0.0476) compared to AdamW (0.0547). This was accompanied by a 12.7% increase in training time.
- **Ablation Study**: Both momentum and Newton-Schulz orthogonalization were observed to be integral to the optimizer's performance. The removal of either component resulted in a significant increase in validation loss. The results also point towards a synergistic relationship between the two components.
- **Hyperparameter Sensitivity**: The optimizer's performance is highly sensitive to the learning rate, with an optimal value found at 0.05 within the tested range. It showed moderate sensitivity to momentum (0.95 optimal) and weak sensitivity to the number of Newton-Schulz steps (7 optimal).

5.2 Future Work

This research is active and continues to evolve. Future experiments planned include:

- A comprehensive learning rate sweep for both Muon and AdamW to visualize their sensitivity curves and identify optimal operating ranges.
- A detailed computational profiling analysis to precisely quantify the overhead of the Newton-Schulz iteration and identify potential optimization bottlenecks.
- Scaling experiments on larger models and datasets to determine if the observed performance characteristics hold true for more complex training regimes.
- Investigation into the interaction between hyperparameters, potentially through a grid search, to find a globally optimal configuration.

Final conclusions will be drawn upon the completion of these and subsequent experiments.

References

[1] Ilya Loshchilov and Frank Hutter. *Decoupled Weight Decay Regularization.* International Conference on Learning Representations (ICLR), 2019. arXiv:1711.05101

[2] Jeremy Bernstein, et al. *On the Distance Between Two Neural Networks and the Stability of Learning.* arXiv:2002.03432, 2020.

[3] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization.* arXiv:1412.6980, 2014.

[4] Noam Shazeer, et al. *Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer.* arXiv:1701.06538, 2017.

[5] Jingyuan Liu, et al. *Muon is Scalable for LLM Training.* arXiv:2502.16982, 2025.

[6] Keller Jordan. *Muon: An optimizer for hidden layers in neural networks.* `https://kellerjordan.github.io/posts/muon/`, 2024.

[7] G. Schulz. *Iterative Berechnung der Reziproken Matrix.* Zeitschrift für Angewandte Mathematik und Mechanik, 13:57–59, 1933.