Muon Outperforms Adam in Tail-End Associative Memory Learning

ICLR 2026 Submission

National University of Singapore, University of Minnesota Sea AI Lab, Yale University

Presenter: Angxiao Yue

December 13, 2025

Outline

Introduction: Muon vs. Adam

- **The Landscape**
  - **Adam**: The standard optimizer for LLMs (optimizes w.r.t. vector ℓ_∞ norm).
  - **Muon**: A novel matrix-parameter optimizer (Jordan et al., 2024).
  - **Impact**: Nearly **2**$\times$ faster than Adam across model sizes.
- **The Motivation (The Missing Link)**
  - Muon performs steepest descent w.r.t. the **Spectral Norm**.
  - *Open Question*: Why does spectral norm optimization outperform ℓ_∞ optimization in Transformers?
  - Current convergence analyses fail to explain this empirical superiority.
- **Core Research Questions**
  1. Which Transformer components benefit most from Muon?
  2. What structural features enable this effective optimization?

Key Insights & Contributions

- **Primary Beneficiaries: Associative Memories**
  - Muon's superiority stems from **Value-Output (VO)** matrices and **FFN** blocks.
  - These components act as the primary *associative memory* stores.
- **Handling Heavy-Tailed Distributions**
  - Real-world data is heavy-tailed (frequent "head" vs. rare "tail" classes).
  - Muon's spectral normalization creates **isotropic** (balanced) weight updates.
  - **Result**: Effectively optimizes **tail classes** without being dominated by head classes.
- **Theoretical Validation**
  - Modeled via one-layer linear associative memory.
  - **Proof**: Muon maintains balanced learning across imbalanced classes; Adam exhibits instability dependent on embedding structure.

Outline

Preliminary: The Muon Optimizer

- **Core Concept**
  - An optimizer tailored for **matrix parameters** (Jordan et al., 2024).
  - Interpreted as steepest descent w.r.t. the **Spectral Norm** (Bernstein & Newhouse, 2024).
  - Produces a scale-invariant update direction by normalizing singular values.
- **Update Rule**
  1. **Momentum Accumulation**:
  $$B_t = \mu B_{t-1} + \nabla_W \mathcal{L}(W_t)$$
  2. **Orthogonalization (Key Step)**:
     - Decompose momentum via SVD: $B_t = U_t S_t V_t^\top$.
     - Discard singular values (S_t) to keep only direction:
     $$O_t = U_t V_t^\top$$
  3. **Parameter Step**:
  $$W_{t+1} = W_t - \eta_t O_t$$
- **Efficient Implementation**: Newton-Schulz Iteration

Preliminary: Transformer Architecture

- **Input Processing**
  - Input sequence of N tokens embedded into $X^{(0)} \in \mathbb{R}^{d \times N}$.
  - Each layer $\ell \in [L]$ consists of an **Attention** module and an **FFN** module.
- **Attention Mechanism**
  - Computes token mixing via heads $h \in [H]$:

$$H^{(\ell)} = X^{(\ell-1)} + \sum_{h=1}^{H} W_{O,h}^{(\ell)} W_{V,h}^{(\ell)} X^{(\ell-1)} \operatorname{sm}\left(A_h^{(\ell)}\right)$$

  - **Roles**:
    - $W_{Q,h}, W_{K,h}$: Capture token relationships (Attention Scores A_h).
    - $W_{V,h}, W_{O,h}$: Apply linear transformations (Content).
- **Feed-Forward Networks (FFN)**
  - Updates representations via non-linear mapping:

$$X^{(\ell)} = H^{(\ell)} + W_{\text{out}}^{(\ell)} \sigma\left(W_{\text{in}}^{(\ell)} H^{(\ell)}\right)$$

  - **Gated Variant**: Includes additional W_{gate} with Hadamard product $\odot$.

Preliminary: Linear Associative Memory (1/2)

- **Definition: Storing Facts as Outer Products**
  - Consider a fact triplet (s, r, o) (Subject, Relation, Object).
  - Maps a key vector e_s (encoding s, r) to a value vector e_o (encoding o).
  - **Memory Construction**: The weight matrix W is the sum of facts:

$$W = \sum_{i=1}^{K} e_{o_i} e_{s_i}^{\top}$$

  - **Retrieval**: $We_{s_i} = e_{o_i}$ (assuming orthogonal keys e_{s_i}).

- **Where Does it Live in Transformers?**
  - **Attention**: The VO matrices (W_V, W_O) act as memory access.
  - **FFN**: The entire block functions as a key-value memory.
  - **Insight**: These components store factual associations learned from pretraining data (e.g., “SpaceX” $\leftrightarrow$ “Elon Musk”).

Preliminary: Optimization Dynamics of Memory (2/2)

- **A Toy Example: The Imbalance Problem**
  - Consider learning two orthogonal facts with frequencies $c_1 \gg c_2$:
    1. F_1: ("France" $\rightarrow$ "Paris") [Frequent/Head]
    2. F_2: ("Italy" $\rightarrow$ "Rome") [Rare/Tail]

- **Gradient Structure**
  - The gradient G scales with data frequency (c_i):

$$G = c_1 \cdot \underbrace{(e_{o_1} e_{s_1}^\top)}_{F_1} + c_2 \cdot \underbrace{(e_{o_2} e_{s_2}^\top)}_{F_2}$$

  - **Standard SGD/Adam**: Updates are dominated by F_1 (magnitude c_1).

- **Muon's Solution (Spectral Normalization)**
  - Muon normalizes singular values, effectively setting $c_1 \approx c_2 \approx 1$.

$$O = UV^\top = 1 \cdot F_1 + 1 \cdot F_2$$

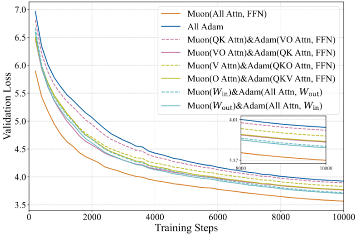
  - **Result**: Learns frequent and rare facts at the **same rate**.

Primary Beneficiaries: Associative Memories (1/4)

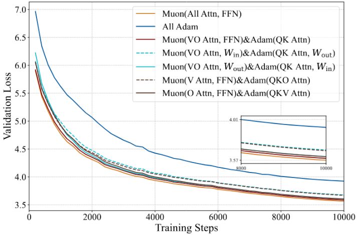
Experimental Setup

- **Model/Data**: 160M NanoGPT trained on FineWeb.
- **Protocol**: Apply Muon to specific components while keeping others on Adam.
  1. *Independent Blocks*: Only one component uses Muon (QK, VO, or FFN).
  2. *Combined Configurations*: Muon on subsets (e.g., VO+FFN) to recover full performance.

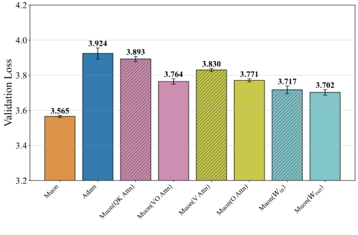
Primary Beneficiaries: Associative Memories (2/4)



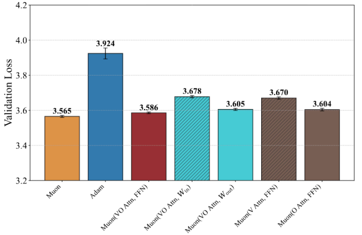
(a) Independent blocks: Val loss over training



(b) Combined configurations: Val loss over training



(c) Independent blocks: Val loss at step 10,000



(d) Combined configurations: Val loss at step 10,000

Primary Beneficiaries: Associative Memories (3/4)

Key Findings

- **Attention**: Muon yields substantially larger gains on **VO weights** (W_V, W_O) than on QK weights (W_Q, W_K).
- **FFN**: All FFN matrices $(W_{\text{in}}, W_{\text{out}}, W_{\text{gate}})$ benefit significantly.
- **Validation**: Applying Muon *only* to **VO + FFN** nearly recovers the full-Muon performance trajectory.

Conclusion (Observation 1)

- Muon is most effective on **associative memory** components (VO + FFN).
- Applying Muon to QK contributes little to overall performance gains.

Primary Beneficiaries: Associative Memories (4/4)

- **The Gradient: Biased by Frequency**
  - Consider learning two orthogonal facts with vastly different frequencies ($c_1 \gg c_2 > 0$):

$$\mathcal{L}(W) = \underbrace{c_1 \|e_{o_1} - We_{s_1}\|^2}_{\text{Head Fact (Frequent)}} + \underbrace{c_2 \|e_{o_2} - We_{s_2}\|^2}_{\text{Tail Fact (Rare)}}$$

  - The gradient G is a weighted sum of outer products (singular values are c_1, c_2):

$$G = \nabla_W \mathcal{L} = \underbrace{c_1 \cdot (e_{o_1} e_{s_1}^\top)}_{\text{Dominant Direction}} + \underbrace{c_2 \cdot (e_{o_2} e_{s_2}^\top)}_{\text{Negligible Direction}}$$

  - **Problem**: Standard optimizers (Adam/SGD) focus almost exclusively on the Head Fact (c_1).
- **Muon Update: Restoring Balance via SVD**
  - Muon performs SVD: $G = U\Sigma V^\top$, where $\Sigma = \mathrm{diag}(c_1, c_2)$.
  - **Normalization**: Muon discards singular values Σ (the frequencies):

$$O = UV^\top = \underbrace{1 \cdot (e_{o_1} e_{s_1}^\top)}_{\text{Head Fact}} + \underbrace{1 \cdot (e_{o_2} e_{s_2}^\top)}_{\text{Tail Fact}}$$

Outline

Handling Heavy-Tailed Distributions (1/6)

- **Singular Energy Distribution**
  - For a weight matrix with singular values $\sigma = (\sigma_1, \dots, \sigma_n)$, define the normalized energy distribution q:

$$q_i = \frac{\sigma_i^2}{\sum_{j=1}^{n} \sigma_j^2}$$

  - Represents the fraction of spectral energy captured by each direction.

- **Metrics for Isotropy (Evenness)**
  1. **Normalized SVD Entropy**:

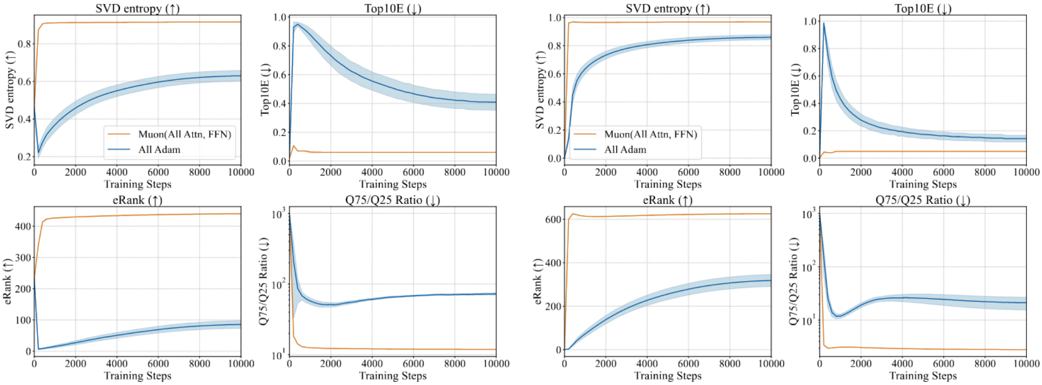
$$H_{\text{norm}}(\sigma) = -\frac{1}{\log n} \sum_{i=1}^{n} q_i \log q_i$$

  2. **Effective Rank**:

$$\mathrm{eRank}(\sigma) = \exp\left(-\sum_{i=1}^{n} q_i \log q_i \right)$$

  3. **Top-k Energy Fraction**: $\mathrm{TopE}_k(\sigma) = \sum_{i=1}^{k} q_i$.
  4. **Eigenvalue Quantile Ratio**: $Q_{75/25}(\sigma) = Q_3(\{\sigma_i^2\}) / Q_1(\{\sigma_i^2\})$.

Handling Heavy-Tailed Distributions (2/6)



(a) VO(Non-gated FFN) (b) W_{out}(Non-gated FFN)

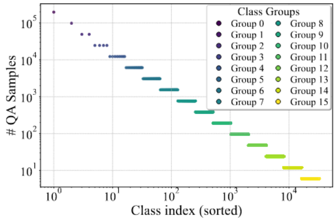
Handling Heavy-Tailed Distributions (3/6)

- **Spectral Dynamics (Averaged over 10 seeds)**
  - **Higher Isotropy**: Muon produces a much more isotropic singular spectrum than Adam throughout training.
  - **Stability**: Muon is robust to random initialization (negligible error bars), whereas Adam is sensitive and fluctuates significantly.
- **Conclusion (Observation 2)**
  - Muon consistently yields weight matrices with **broadly distributed spectral energy**.
  - Result: Supports **richer feature representations** in associative memory components.

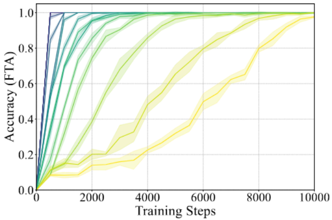
Handling Heavy-Tailed Distributions (4/6)

- **Task Overview**
  - **Goal**: Evaluate how well optimizers learn associative memories under data imbalance.
  - **Dataset**: Synthetic QA dataset containing biographical facts (e.g., birthday, company) for $> 200,000$ individuals (Allen-Zhu & Li, 2024).
- **Key Characteristic: Heavy-Tailed Distribution**
  - Frequencies of individuals follow a **Power-Law distribution**.
  - **Simulation**: Mimics real-world knowledge where a few “head” entities are frequent, but the vast majority are “tail” (rare).
- **Model & Evaluation**
  - **Architecture**: 160M NanoGPT.
  - **Metric**: **First Token Accuracy (FTA)** on the answers.
  - **Baselines**: Comparing **Muon**, **Adam**, and **SGD**.

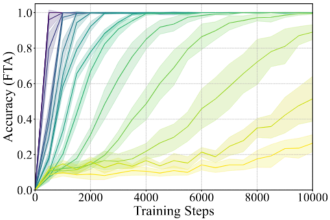
Handling Heavy-Tailed Distributions (5/6)



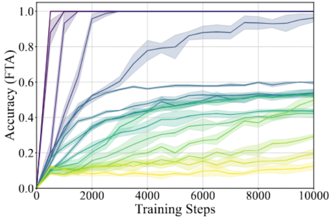
(a) Sample/class



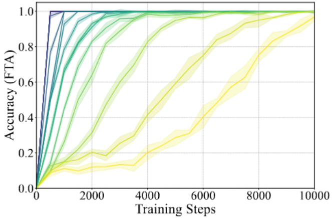
(b) Muon



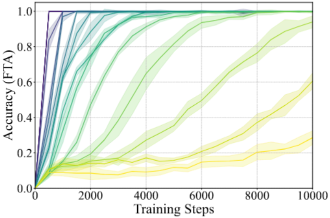
(c) Adam



(d) SGD+Momentum



(e) Muon(VO,FFN)/Adam(QK)



(f) Muon(QK)/Adam(VO,FFN)

Handling Heavy-Tailed Distributions (6/6)

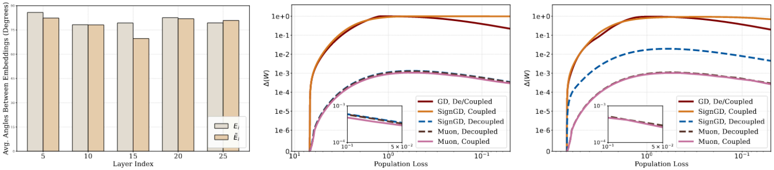
- **Performance on Heavy-Tailed Data**
  - **Head Classes**: Muon matches Adam's strong performance.
  - **Tail Classes**: Muon substantially outperforms Adam, achieving faster convergence and narrowing the head-tail gap.
  - **Stability**: Muon exhibits consistently tighter error bars (lower variance) compared to Adam.
- **Source of Improvement**
  - Hybrid experiments confirm **VO+FFN** are the primary drivers.
  - Applying Muon only to **QK** yields limited improvement.
- **Control Task: In-Context Linear Regression**
  - A task primarily dependent on **QK parameters**.
  - **Result**: Muon performs similarly to Adam.
  - *Implication*: Confirms Muon's superiority is specific to associative memory components, not general optimization.
- **Conclusion (Observation 3)**: In knowledge-intensive tasks, Muon effectively narrows the performance gap between frequent and rare classes.

Theoretical Validation (1/3)

- **Experimental Setup**
  - **Task**: One-layer associative memory model under class imbalance.
  - **Optimizers**: GD, SignGD (proxy for Adam), and Muon.
  - **Embeddings**: Tested two regimes:
    1. *Support-Decoupled*: Disjoint indices (orthogonal-like).
    2. *Support-Coupled*: Overlapping supports (feature interference).
- **Metric: Maximal Probability Gap**
  - Quantifies the disparity between the best-learned and worst-learned items:

$$\Delta(W) := \max_{i,j \in [K]} \left([f_W(E_i)]_i - [f_W(E_j)]_j \right)$$

  - A larger $\Delta(W)$ indicates greater **learning imbalance** (Head vs. Tail gap).



(a) Average Angles Between $E_i/\widetilde{E}_i$ (b) One-step Optimization Results (c) Multi-step Optimization Results

Theoretical Validation (2/3)

- **The Setup**
  - We analyze a one-step update from initialization $W_0 = 0$.
  - We choose a step size η such that the *best-learned* class reaches probability $1 - \epsilon$.
- **The Metric: Infimum Correct-Class Probability ($\varrho^{\epsilon}_{\mathbf{opt}}$)**
  - We measure the performance of the *worst-learned* class at that same step η:

$$\varrho^{\epsilon}_{\text{opt}} = \inf_{\eta \geq 0} \left\{ \min_k [f_{W_\eta}(E_k)]_k \,\middle|\, \max_k [f_{W_\eta}(E_k)]_k \geq 1 - \epsilon \right\}$$

  - **Interpretation**:
    - If $\varrho^{\epsilon}_{\text{opt}} \approx 1 - \epsilon$: **Balanced Learning** (Tail $\approx$ Head).
    - If $\varrho^{\epsilon}_{\text{opt}} \approx 0$: **Imbalanced Learning** (Tail lags behind).
- **Data Imbalance Ratio (r)**
  - Let $r := \frac{\min_k p_k}{\max_k p_k} \in (0, 1]$. Small r implies heavy-tailed distribution.

Theoretical Validation (3/3)

- **Gradient Descent (GD): Sensitive to Imbalance**
  - For any embeddings satisfying assumptions:

$$\varrho^{\epsilon}_{\mathrm{GD}} = O\left(\epsilon^{-r} K^{r-1}\right)$$

  - **Insight**: Performance heavily depends on imbalance ratio r.
  - If data is heavy-tailed ($r \ll 1$), $\varrho^{\epsilon}_{\mathrm{GD}} \to 0$. **GD fails on tail classes.**

- **Muon: Consistently Balanced**
  - For *any* valid embeddings, Muon achieves:

$$\varrho^{\epsilon}_{\mathrm{Muon}} \geq 1 - \epsilon\left(1 + O\left(\frac{\log K}{K}\right)\right)$$

  - **Insight**: The bound is **independent** of r. Muon learns tail classes as effectively as head classes.
  - **Mechanism**: The update matrix aligns with associative memory structure:

$$G_{\mathrm{Muon}}(W_0) \approx -\tilde{E}E^{\top} = -\sum_k \tilde{E}_k E_k^{\top}$$

Outline

Conclusion

- **Decoded Muon's Success**
  - Muon is not a generic accelerator; it specifically targets **Associative Memory** components (**VO** attention matrices and **FFN** blocks).
- **The Core Mechanism**
  - The Muon update rule aligns perfectly with the **outer-product structure** of linear associative memories.
  - By normalizing singular values, Muon generates **isotropic updates**, preventing spectral energy from concentrating only on dominant directions.
- **Impact on Heavy-Tailed Learning**
  - Real-world data is heavily imbalanced (Head vs. Tail).
  - Muon enables **balanced learning**: it matches Adam on frequent concepts while substantially improving performance on **rare (tail) concepts**.
- **Future Direction**: Extending this spectral normalization intuition from matrices to higher-order tensor products.

Thank you!