Muon Outperforms Adam in Tail-End Associative Memory Learning

Shuche Wang* **Fengzhuo Zhang*** Jiaxiang Li* Cunxiao Du Chao Du Tianyu Pang
Zhuoran Yang Mingyi Hong Vincent Y. F. Tan

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

ASAP Seminar

Outline

- **Problem**: Why and how Muon optimizes LLMs faster than Adam?
- **TL;DR**: Muon is more aligned with the associative memory structure in transformers.
  - Muon is most effective when applied to VO and FFNs (associative memory parameters).
  - Muon consistently learns more isotropic parameters than Adam.
  - Muon learns the tail classes better than Adam for long-tailed distributions.

Background

Adam

Adam Iteration for a Matrix $W \in \mathbb{R}^{m\times n}$

- (First and second moments est.) $g_t = \mathtt{vec}(\nabla_W L(W_t))$

$$m_t = \beta_1 m_{t-1} + (1-\beta_1)g_t, \quad v_t = \beta_2 v_{t-1} + (1-\beta_2)g_t^2$$
$$\hat{m}_t = m_t/(1-\beta_1^t), \quad \hat{v}_t = v_t/(1-\beta_2^t)$$

- (Element-wise normalized update)

$$\mathtt{vec}(W_{t+1}) = \mathtt{vec}(W_t) - \eta_t\, \hat{m}_t/(\sqrt{\hat{v}_t} + \varepsilon)$$

Adam

Adam Iteration for a Matrix $W \in \mathbb{R}^{m \times n}$

- (First and second moments est.) $g_t = \mathtt{vec}(\nabla_W L(W_t))$

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t, \quad v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2$$
$$\hat{m}_t = m_t/(1-\beta_1^t), \quad \hat{v}_t = v_t/(1-\beta_2^t)$$

- (Element-wise normalized update)

$$\mathtt{vec}(W_{t+1}) = \mathtt{vec}(W_t) - \eta_t \, \hat{m}_t/(\sqrt{\hat{v}_t} + \varepsilon)$$

Intuition: Adam updates parameters with element-wise normalized gradients.

Muon

Muon Iteration for a Matrix $W \in \mathbb{R}^{m\times n}$

- (First moment est.) $G_t = \nabla_W L(W_t)$

$$B_t = \mu B_{t-1} + G_t$$

- (Spectral-wise normalized update)

$$B_t = U_t \Sigma_t V_t^\top \,\textbf{(SVD)}, \quad O_t = U_t V_t^\top$$
$$W_{t+1} = W_t - \eta_t O_t$$

Muon

Muon Iteration for a Matrix $W \in \mathbb{R}^{m\times n}$

- (First moment est.) $G_t = \nabla_W L(W_t)$

$$B_t = \mu B_{t-1} + G_t$$

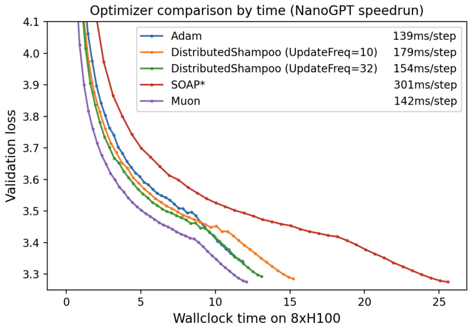
- (Spectral-wise normalized update)

$$B_t = U_t \Sigma_t V_t^\top \textbf{(SVD)}, \quad O_t = U_t V_t^\top$$
$$W_{t+1} = W_t - \eta_t O_t$$

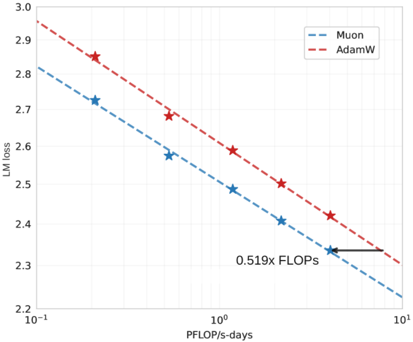
Intuition: Muon updates parameters with spectral-wise normalized gradients.

Practical Efficiency of Adam and Muon

Muon is much **faster** than Adam across a wide range of model sizes and architectures.



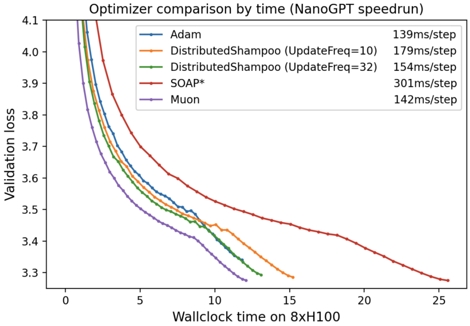
(a) Transformer results (Jordan et al., 2024).



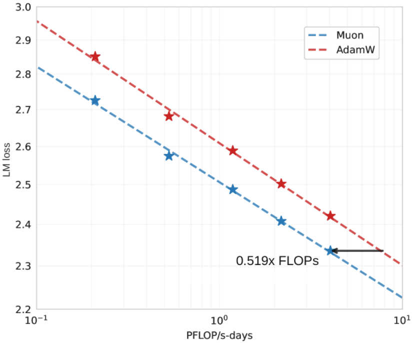
(b) MoE results (Kimi, 2025).

Practical Efficiency of Adam and Muon

Muon is much **faster** than Adam across a wide range of model sizes and architectures.



(c) Transformer results (Jordan et al., 2024).



(d) MoE results (Kimi, 2025).

Why and how is Muon faster than Adam?

Empirical findings

Observation 1: VO & FFN are the primary beneficiaries

Practical Implementations of Muon for Transformers

```
For parameter W in Transformer:
  If (W is token emb. or lm_head) or (W is 1-dim):
    Implement Adam on W.
  Else:
    Implement Muon on W.
```

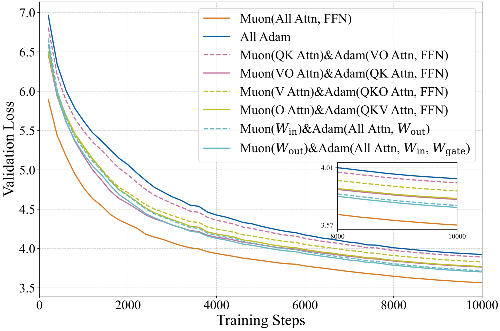
Question

Are all parameters contributing equally to Muon's superiority? Which transformer components benefit most from Muon compared to Adam?

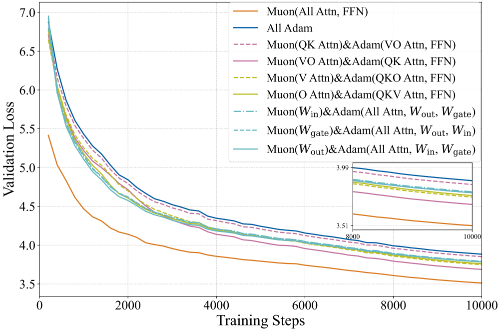
Observation 1: VO & FFN are the primary beneficiaries

- 160M NanoGPT on *FineWeb*; compare **independent** vs. **combined** Muon ablations.
  - **Independent:** Muon in QK, VO, V, O, W_{in}, W_{out}, W_{gate}, respectively.
  - **Combined:** Muon in VO+FFN, VO+W_{in}, VO+W_{out}, V+FFN, O+FFN.
- Attention $\{W_Q, W_K, W_V, W_O\}$ and FFN $\{W_{\mathrm{in}}, W_{\mathrm{out}}, W_{\mathrm{gate}}\}$.

Observation 1: VO & FFN are the primary beneficiaries

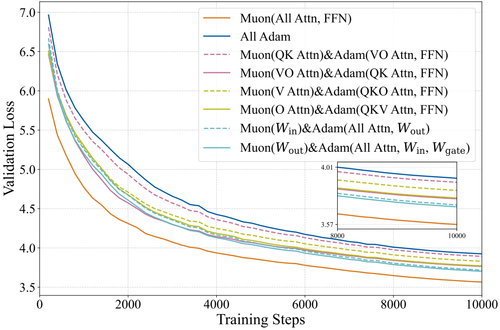


(a) Independent Blocks with Non-gated FFN

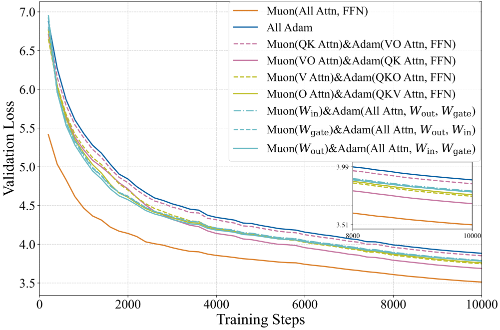


(b) Independent Blocks with Gated FFN

Observation 1: VO & FFN are the primary beneficiaries



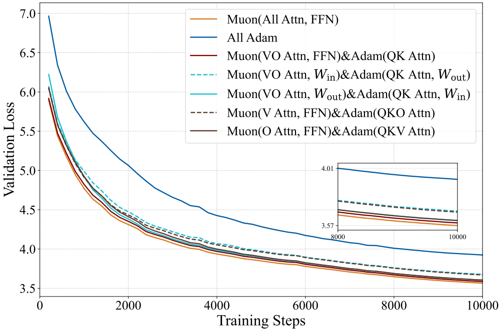
(a) Independent Blocks with Non-gated FFN



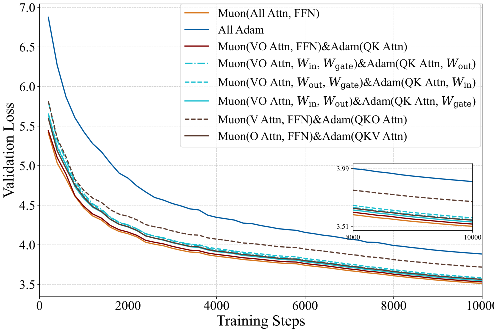
(b) Independent Blocks with Gated FFN

- **VO** weights show larger gains under Muon than the **QK** weights.
- Only W_V or only W_O already yields much larger gains than applying it to **QK**.
- W_{in}, W_{gate}, and W_{out} all benefit from Muon.

Observation 1: VO & FFN are the primary beneficiaries

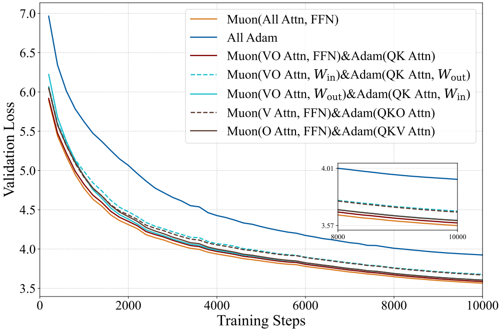


(c) Combined Configuration with Non-gated FFN

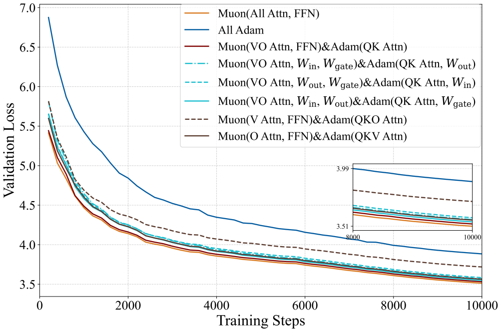


(d) Combined Configuration with Gated FFN

Observation 1: VO & FFN are the primary beneficiaries



(c) Combined Configuration with Non-gated FFN



(d) Combined Configuration with Gated FFN

- **VO+FFN** under Muon **nearly matches full-Muon**; **QK** gains are small.
- Within VO, W_O is more influential than W_V (clearer in non-gated FFN).

Observation 1: The Findings

Observation Summary

Muon is most effective when applied to VO and FFN; in particular, applying Muon to only **VO+FFN** almost recovers the **full-Muon** trajectory.

Why VO & FFN benefit: An Associative Memory View

Question

What structural features of the transformer allow Muon to optimize these components more effectively?

Background: Transformers as Associative Memories

Linear Associative Memories

Certain weight matrices in LLMs function as **linear associative memories**, mapping keys to values to store and retrieve knowledge (Bietti et al., 2023; Meng et al., 2022).

- Key: subject-relation embedding $\mathbf{e}_s \leftarrow (s = \text{subject}, r = \text{relation})$
- Value: object embedding $\mathbf{e}_o \leftarrow (o = \text{object})$

- s is the subject, r the relation, and o the object (e.g., $s =$"The United Nations headquarters", $r =$"is located in", $o =$"New York City")

- **Retrieval:** A weight matrix W recalls the object from the subject-relation key.

$$\mathbf{e}_o = W\mathbf{e}_s$$

- **Construction:** W is formed by summing the outer products of key-value pairs.

$$W = \sum_{i=1}^{K} \mathbf{e}_{o_i}\mathbf{e}_{s_i}^\top$$

Example: Storing two facts in $\mathbb{R}^2$

Let's store: **(grass, green)** and **(sky, blue)**

Background: Transformers as Associative Memories

- **Retrieval:** A weight matrix W recalls the object from the subject-relation key.

$$\mathbf{e}_o = W\mathbf{e}_s$$

- **Construction:** W is formed by summing the outer products of key-value pairs.

$$W = \sum_{i=1}^{K} \mathbf{e}_{o_i}\mathbf{e}_{s_i}^\top$$

Example: Storing two facts in $\mathbb{R}^2$

Let's store: **(grass, green)** and **(sky, blue)**

- **1. Assign Orthogonal Embeddings:**
  - Keys: $\mathbf{e}_{\text{grass}} = (1, 0)^\top$, $\mathbf{e}_{\text{sky}} = (0, 1)^\top$
  - Values: $\mathbf{e}_{\text{green}} = (1, 0)^\top$, $\mathbf{e}_{\text{blue}} = (0, 1)^\top$

Background: Transformers as Associative Memories

- **Retrieval:** A weight matrix W recalls the object from the subject-relation key.

$$\mathbf{e}_o = W\mathbf{e}_s$$

- **Construction:** W is formed by summing the outer products of key-value pairs.

$$W = \sum_{i=1}^{K} \mathbf{e}_{o_i}\mathbf{e}_{s_i}^\top$$

Example: Storing two facts in $\mathbb{R}^2$

Let's store: **(grass, green)** and **(sky, blue)**

- **1. Assign Orthogonal Embeddings:**
  - Keys: $\mathbf{e}_{\text{grass}} = (1,0)^\top$, $\mathbf{e}_{\text{sky}} = (0,1)^\top$
  - Values: $\mathbf{e}_{\text{green}} = (1,0)^\top$, $\mathbf{e}_{\text{blue}} = (0,1)^\top$
- **2. Construct Memory Matrix W:**

$$W = \mathbf{e}_{\text{green}}\mathbf{e}_{\text{grass}}^\top + \mathbf{e}_{\text{blue}}\mathbf{e}_{\text{sky}}^\top = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Background: Transformers as Associative Memories

- **Retrieval:** A weight matrix W recalls the object from the subject-relation key.

$$\mathbf{e}_o = W\mathbf{e}_s$$

- **Construction:** W is formed by summing the outer products of key-value pairs.

$$W = \sum_{i=1}^{K} \mathbf{e}_{o_i}\mathbf{e}_{s_i}^\top$$

Example: Storing two facts in $\mathbb{R}^2$

Let's store: **(grass, green)** and **(sky, blue)**

- **1. Assign Orthogonal Embeddings:**
  - Keys: $\mathbf{e}_{\text{grass}} = (1,0)^\top$, $\mathbf{e}_{\text{sky}} = (0,1)^\top$
  - Values: $\mathbf{e}_{\text{green}} = (1,0)^\top$, $\mathbf{e}_{\text{blue}} = (0,1)^\top$
- **2. Construct Memory Matrix W:**

$$W = \mathbf{e}_{\text{green}}\mathbf{e}_{\text{grass}}^\top + \mathbf{e}_{\text{blue}}\mathbf{e}_{\text{sky}}^\top = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

- **3. Retrieve Facts:**
  - Query for "grass":

$$W\mathbf{e}_{\text{grass}} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \mathbf{e}_{\text{green}}$$

Why VO & FFN benefit: An Associative Memory View

VO and FFN behave as linear associative memories. (Bietti et al., 2023; Meng et al., 2022)

$$\underbrace{W}_{\text{VO or FFN}} \approx \sum_{i=1}^{K} \underbrace{e_{o,i}}_{\text{value}} \underbrace{e_{s,i}^{\top}}_{\text{key}} \quad \text{(outer-product store)}$$

Why VO & FFN benefit: An Associative Memory View

VO and FFN behave as linear associative memories. (Bietti et al., 2023; Meng et al., 2022)

$$\underbrace{W}_{\text{VO or FFN}} \approx \sum_{i=1}^{K} \underbrace{e_{o,i}}_{\text{value}} \underbrace{e_{s,i}^{\top}}_{\text{key}} \quad \text{(outer-product store)}$$

Muon step:

$$G = USV^{\top} = \sum_{i=1}^{d} s_i u_i v_i^{\top} \quad \Rightarrow \quad O = UV^{\top} = \sum_{i=1}^{d} u_i v_i^{\top}$$

updates all ***orthogonal facts*** *at the same rate*

Verifying the Insight: Two Perspectives

- **Weight Spectra:**
  - Weight matrices learned with **Muon** exhibit a more **isotropic singular-value spectrum** than those learned with **Adam**.

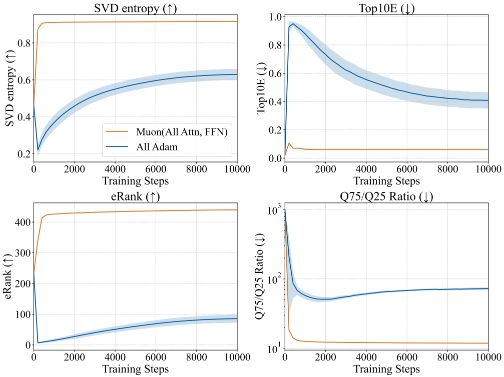
Verifying the Insight: Two Perspectives

- **Weight Spectra:**
  - Weight matrices learned with **Muon** exhibit a more **isotropic singular-value spectrum** than those learned with **Adam**.
- **Knowledge Acquisition:**
  - **Muon** yields more balanced learning across entities and frequencies (head and tail) than **Adam**.

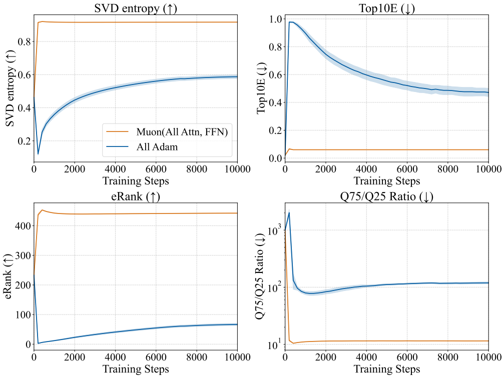
Observation 2: Muon learns more isotropic spectra

- Metrics: Based on the singular energy distribution $q_i = \sigma_i^2 / \sum \sigma_j^2$
  - **SVD entropy**: $H_{\text{norm}} = -\frac{1}{\log n} \sum_{i=1}^{n} q_i \log q_i$;
  - **effective rank**: $\text{eRank} = \exp\left(-\sum q_i \log q_i\right)$;
  - **Top-k energy**: $\text{TopE}_k = \frac{\sum_{i=1}^{k} \sigma_i^2}{\sum_{j=1}^{n} \sigma_j^2}$;
  - Q_{75}/Q_{25}: $Q_{75/25} = \frac{Q_3(\{\sigma_i^2\})}{Q_1(\{\sigma_i^2\})}$.

Observation 2: Muon learns more isotropic spectra

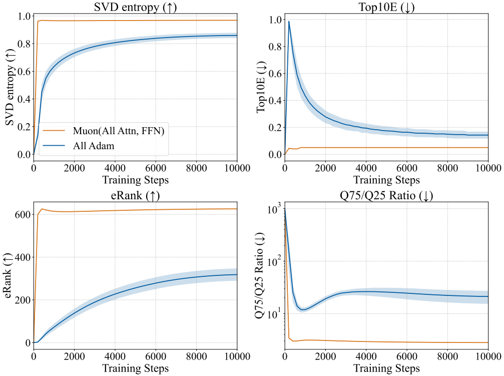


(a) VO(Non-gated FFN)

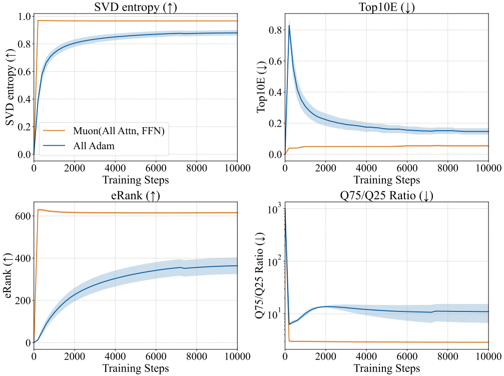


(b) VO(Gated FFN)

Observation 2: Muon learns more isotropic spectra

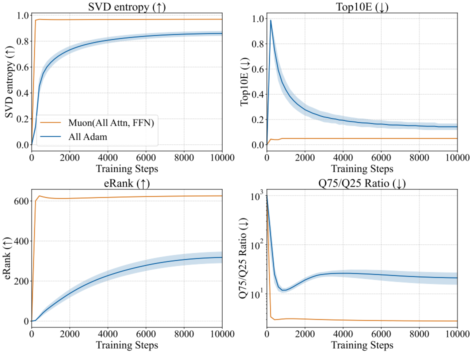


(c) W_{out}(Non-gated FFN)

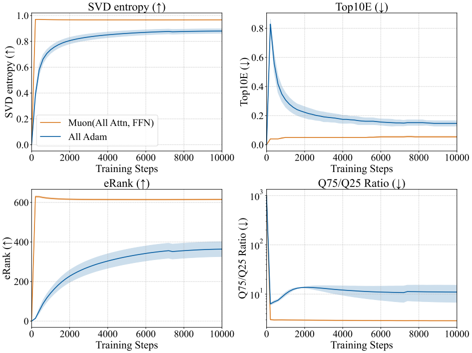


(d) W_{out}(Gated FFN)

Observation 2: Muon learns more isotropic spectra



(c) W_{out}(Non-gated FFN)



(d) W_{out}(Gated FFN)

- From early training, Muon yields **more isotropic** spectra and **seed-stable** curves; Adam **fluctuates**.
- Effects are consistent on VO and FFN (gated and non-gated FFN).

Observation 2: Findings

Observation Summary

Muon consistently yields more **isotropic weight matrices** with broadly distributed spectral energy than **Adam**, both throughout training and across random initializations, thereby supporting **richer feature representations**.

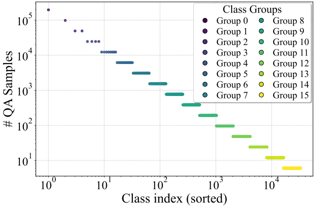
Observation 3: Muon Acquires Knowledge More Evenly Compared To Adam

- Synthetic QA dataset containing biographical information (e.g., name, birthday, and company) for over 200,000 individuals (Allen-Zhu and Li, 2024); **power-law** sampling across classes;

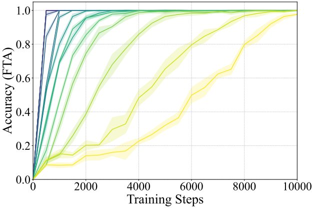
  > ***Ashton Hilda Older*** *has a birthday that falls on* ***February 01, 2063****.* ***Miami, FL*** *is the birthplace of he. He is an alumnus of* ***Saddleback College****. He has a* ***General Literature*** *education. He works closely with* ***BlockFi****. For professional growth, he chose to relocate to* ***Jersey City****.*

- Metric: First-Token Accuracy (FTA).

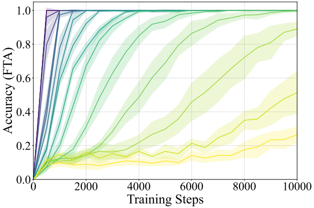
Observation 3: Muon Acquires Knowledge More Evenly Compared To Adam



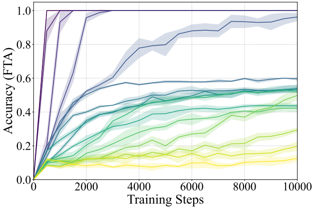
(a) Sample/class



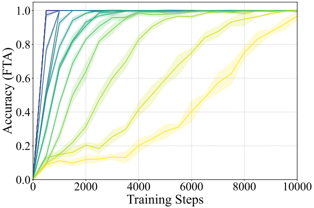
(b) Muon



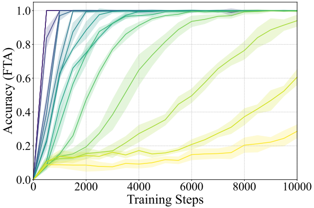
(c) Adam



(d) SGD+Momentum

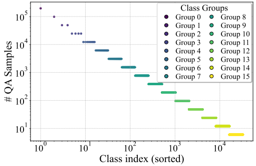


(e) Muon(VO,FFN),Adam(QK)

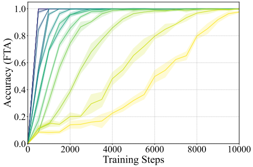


(f) Muon(QK),Adam(VO,FFN)

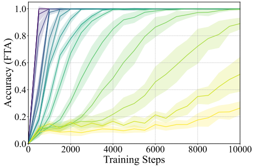
Observation 3: Muon Acquires Knowledge More Evenly Compared To Adam



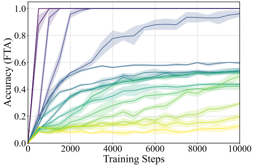
(a) Sample/class



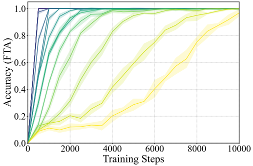
(b) Muon



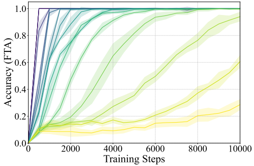
(c) Adam



(d) SGD+Momentum



(e) Muon(VO,FFN)



(f) Muon(QK)

- **Head**: all optimizers do well; **Tail**: **Muon** $\gg$ **Adam** with **lower variance**.
- Muon on **VO+FFN** retains most gains; **QK-only** improvement is limited.

Observation 3: Findings

Observation Summary

In **heavy-tailed, knowledge-intensive** tasks, Muon matches Adam's strong performance in the head classes while substantially improving learning on **tail classes**, narrowing the **head-tail gap** and accelerating convergence.

Case Study of One-Layer Models

Setup

- Learn K triplets $\{(s_i, r_i, o_i)\}_{i=1}^{K}$.
- Embed (s_i, r_i) to $E_i \in \mathbb{R}^d$ and o_i to $\tilde{E}_i \in \mathbb{R}^d$ for $i \in [K]$.

Setup

- Learn K triplets $\{(s_i, r_i, o_i)\}_{i=1}^K$.
- Embed (s_i, r_i) to $E_i \in \mathbb{R}^d$ and o_i to $\tilde{E}_i \in \mathbb{R}^d$ for $i \in [K]$.
- Queried by (s_i, r_i), the network outputs the probability over $\{o_i\}_{i=1}^K$ as

$$f_W(E_k) = \mathtt{sm}(\tilde{E}^\top W E_k) \in \mathbb{R}^K,$$

where $E = [E_1, \cdots, E_K] \in \mathbb{R}^{d \times K}$, and $\tilde{E} = [\tilde{E}_1, \cdots, \tilde{E}_K] \in \mathbb{R}^{d \times K}$.

Setup

- Learn K triplets $\{(s_i, r_i, o_i)\}_{i=1}^K$.
- Embed (s_i, r_i) to $E_i \in \mathbb{R}^d$ and o_i to $\tilde{E}_i \in \mathbb{R}^d$ for $i \in [K]$.
- Queried by (s_i, r_i), the network outputs the probability over $\{o_i\}_{i=1}^K$ as

$$f_W(E_k) = \mathtt{sm}(\tilde{E}^\top W E_k) \in \mathbb{R}^K,$$

where $E = [E_1, \cdots, E_K] \in \mathbb{R}^{d \times K}$, and $\tilde{E} = [\tilde{E}_1, \cdots, \tilde{E}_K] \in \mathbb{R}^{d \times K}$.

- The network is trained with loss

$$\mathcal{L}(W) = -\sum_{k=1}^{K} p_k \log[f_W(E_k)]_k,$$

where p_k is the frequency or probability of the k-th triplet.

Setup

Assumption

The data frequency imbalance is modeled by two groups, where $p_k = \alpha/L$ for $k \in [L]$ and $p_k = (1-\alpha)/(K-L)$ for $k > L$.

Define $\beta = L/K$. The dataset is perfectly **balanced** when $\alpha = \beta$, and becomes highly **imbalanced** when $\alpha \ll \beta$ or $\beta \ll \alpha$.

Setup

Assumption

The data frequency imbalance is modeled by two groups, where $p_k = \alpha/L$ for $k \in [L]$ and $p_k = (1-\alpha)/(K-L)$ for $k > L$.

Define $\beta = L/K$. The dataset is perfectly **balanced** when $\alpha = \beta$, and becomes highly **imbalanced** when $\alpha \ll \beta$ or $\beta \ll \alpha$.

Assumption

The embeddings E and $\tilde{E}$ are orthonormal, i.e., $E^\top E = \tilde{E}^\top \tilde{E} = I_{K,K}$.

If embeddings are not normalized, this becomes another source of imbalance that **interacts** with the data frequency imbalance.

Setup

- **Vanilla GD**:

$$W_{t+1}^{\mathsf{GD}} = W_t^{\mathsf{GD}} - \eta_{t+1} \nabla_W \mathcal{L}(W_t^{\mathsf{GD}}).$$

- **Adam with $\beta_1 = 0, \beta_2 = 0$ (SignGD)**:

$$W_{t+1}^{\mathsf{SignGD}} = W_t^{\mathsf{SignGD}} - \eta_{t+1} \operatorname{sign}\left(\nabla_W \mathcal{L}(W_t^{\mathsf{SignGD}})\right).$$

- **Muon with $\mu = 0$**:

$$W_{t+1}^{\mathsf{Muon}} = W_t^{\mathsf{Muon}} - \eta_{t+1} U_t \mathsf{norm}(\Sigma_t) V_t^\top,$$

  where $\mathsf{norm}(\cdot)$ normalizes all non-zero elements to 1 (element-wise), and $\nabla_W \mathcal{L}(W_t^{\mathsf{Muon}}) = U_t \Sigma_t V_t^\top$ is the SVD of the gradient.

Experimental Results

Intuitions recall:

Muon is **spectral-wise** normalized, while Adam is **element-wise** normalized.

Experimental Results

Intuitions recall:

Muon is **spectral-wise** normalized, while Adam is **element-wise** normalized.

Embedding settings:

- Support-decoupled: $\mathrm{Supp}(E_i)/\mathrm{Supp}(\tilde{E}_i)$ are **disjoint** for different i, e.g., one-hot bases.
- Support-coupled: Supports may **overlap**, e.g., general unitary matrix.

Experimental Results

Intuitions recall:

Muon is **spectral-wise** normalized, while Adam is **element-wise** normalized.

Embedding settings:

- Support-decoupled: $\mathrm{Supp}(E_i)/\mathrm{Supp}(\tilde{E}_i)$ are **disjoint** for different i, e.g., one-hot bases.
- Support-coupled: Supports may **overlap**, e.g., general unitary matrix.

Optimizer settings:

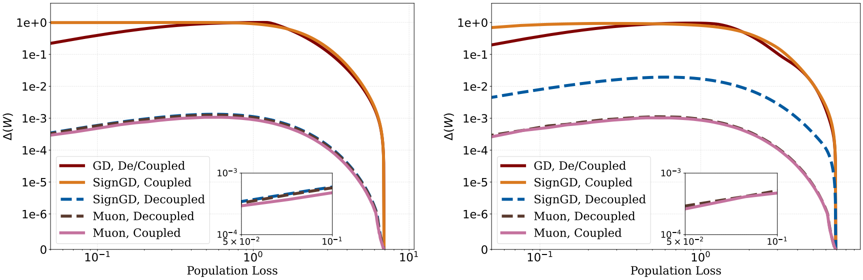
- One-step: Take a single update with a scaled step size.
- Multi-step: Take multiple updates.

Experimental Results

Imbalance metric: *maximal probability gap* $\Delta(W) = \max_{i,j\in[K]}[f_W(E_i)]_i - [f_W(E_j)]_j$,

Experimental Results

Imbalance metric: *maximal probability gap* $\Delta(W) = \max_{i,j\in[K]} [f_W(E_i)]_i - [f_W(E_j)]_j$,



(a) One-step Optimization Results

(b) Multi-step Optimization Results

- Muon **consistently** achieves balanced learning across all items for any embeddings.
- Adam's ability to learn balanced items depends on the **properties of the embeddings**.

Theoretical Understanding

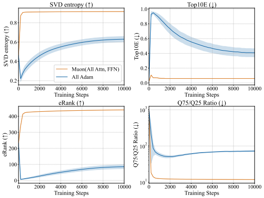
Theorem (Informal)

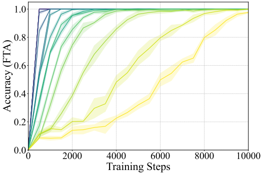
- **Muon:** near-isotropic updates $\Rightarrow$ worst-class prob $\gtrsim 1 - \varepsilon\Big(1 + \mathcal{O}\big(\frac{\log K}{K}\big)\Big)$ once the best class hits $1 - \varepsilon$. Imbalance: $\tilde{O}(\varepsilon \log K / K)$

Theoretical Understanding

Theorem (Informal)

- **Muon:** near-isotropic updates $\Rightarrow$ worst-class prob $\gtrsim 1 - \varepsilon\left(1 + \mathcal{O}\left(\frac{\log K}{K}\right)\right)$ once the best class hits $1 - \varepsilon$. Imbalance: $\tilde{O}(\varepsilon \log K / K)$





Theoretical Understanding

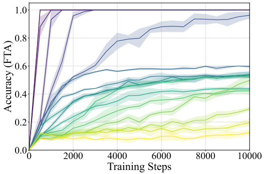
Theorem (Informal)

- **Muon:** near-isotropic updates $\Rightarrow$ worst-class prob $\gtrsim 1 - \varepsilon\Big(1 + \mathcal{O}\big(\frac{\log K}{K}\big)\Big)$ once the best class hits $1 - \varepsilon$. Imbalance: $\tilde{O}(\varepsilon \log K / K)$
- **GD:** imbalance controlled by $r(\alpha, \beta) < 1$; worst-class prob can be $O(\varepsilon^{-r} K^{r-1})$. Imbalance: $1 - \varepsilon - O(\varepsilon^{-r} K^{r-1})$

Theoretical Understanding

Theorem (Informal)

- **Muon:** near-isotropic updates $\Rightarrow$ worst-class prob $\gtrsim 1-\varepsilon\left(1+\mathcal{O}\left(\frac{\log K}{K}\right)\right)$ once the best class hits $1-\varepsilon$. Imbalance: $\tilde{O}(\varepsilon \log K/K)$
- **GD:** imbalance controlled by $r(\alpha,\beta)<1$; worst-class prob can be $O(\varepsilon^{-r}K^{r-1})$. Imbalance: $1-\varepsilon-O(\varepsilon^{-r}K^{r-1})$



Theoretical Understanding

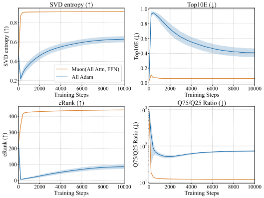
Theorem (Informal)

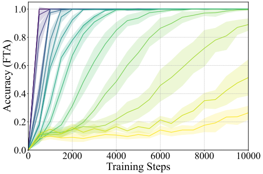
- **Muon:** near-isotropic updates $\Rightarrow$ worst-class prob $\gtrsim 1 - \varepsilon\Big(1 + \mathcal{O}\big(\frac{\log K}{K}\big)\Big)$ once the best class hits $1 - \varepsilon$. Imbalance: $\tilde{O}(\varepsilon \log K / K)$
- **GD:** imbalance controlled by $r(\alpha, \beta) < 1$; worst-class prob can be $O(\varepsilon^{-r} K^{r-1})$. Imbalance: $1 - \varepsilon - O(\varepsilon^{-r} K^{r-1})$
- **Adam:** embedding-dependent; can match Muon (disjoint supports) or degrade (overlap), with spectral decay.

Theoretical Understanding

Theorem (Informal)

- **Muon:** near-isotropic updates $\Rightarrow$ worst-class prob $\gtrsim 1 - \varepsilon\Big(1 + \mathcal{O}\big(\frac{\log K}{K}\big)\Big)$ once the best class hits $1 - \varepsilon$. Imbalance: $\tilde{O}(\varepsilon \log K / K)$
- **GD:** imbalance controlled by $r(\alpha, \beta) < 1$; worst-class prob can be $O(\varepsilon^{-r} K^{r-1})$. Imbalance: $1 - \varepsilon - O(\varepsilon^{-r} K^{r-1})$
- **Adam:** embedding-dependent; can match Muon (disjoint supports) or degrade (overlap), with spectral decay.





Why aligned with associative memory? (intuition)

- Gradient $G = U\Sigma V^\top = \sum_i \sigma_i u_i v_i^\top$; Muon uses $UV^\top = \sum_i u_i v_i^\top$: **equal-magnitude** updates per orthogonal outer product.
- Linear memory $W = \sum_i e_{o_i} e_{s_i}^\top$; singular values encode frequency; Muon **equalizes** learning rates across head/tail facts.
- Adam's elementwise normalization may **disrupt** matrix structure $\Rightarrow$ imbalance & spectral concentration.

Summary

Takeaway

Muon's spectral-normalized updates align with the outer-product form of linear associative memory, delivering isotropic spectra and balanced tail learning. **VO+FFN** is the main battleground.

Relationship to Other Understandings and Phenomena

- Bernstein and Newhouse (2024): Muon and Adam are steepest gradient descents with respect to the operator norm and vector inf norm, respectively.
  $\rightarrow$ One way to explain why the operator norm is better than the vector inf norm.

Relationship to Other Understandings and Phenomena

- Bernstein and Newhouse (2024): Muon and Adam are steepest gradient descents with respect to the operator norm and vector inf norm, respectively.
  $\rightarrow$ One way to explain why the operator norm is better than the vector inf norm.
- Zhang et al. (2025): Adopt Muon for online memory updating in RNNs.
  $\rightarrow$ Another evidence to support Muon's superiority in memory structure.

Relationship to Other Understandings and Phenomena

- Bernstein and Newhouse (2024): Muon and Adam are steepest gradient descents with respect to the operator norm and vector inf norm, respectively.
  $\rightarrow$ One way to explain why the operator norm is better than the vector inf norm.
- Zhang et al. (2025): Adopt Muon for online memory updating in RNNs.
  $\rightarrow$ Another evidence to support Muon's superiority in memory structure.
- The exploration and exploitation view in ZhiHu
  $\rightarrow$ Effective learning on the tail classes can be viewed as exploration.

Thanks for listening!

References

Allen-Zhu, Z. and Li, Y. (2024). Physics of language models: Part 3.3, knowledge capacity scaling laws. *arXiv preprint arXiv:2404.05405*.

Bernstein, J. and Newhouse, L. (2024). Old optimizer, new norm: An anthology. *arXiv preprint arXiv:2409.20325*.

Bietti, A., Cabannes, V., Bouchacourt, D., Jegou, H., and Bottou, L. (2023). Birth of a transformer: A memory viewpoint. *Advances in Neural Information Processing Systems*, 36:1560–1588.

Jordan, K., Jin, Y., Boza, V., Jiacheng, Y., Cecista, F., Newhouse, L., and Bernstein, J. (2024). Muon: An optimizer for hidden layers in neural networks, 2024. *URL https://kellerjordan. github. io/posts/muon*, 6.

Kimi, T. (2025). Kimi k2: Open agentic intelligence. *arXiv preprint arXiv:2507.20534*.

Meng, K., Bau, D., Andonian, A., and Belinkov, Y. (2022). Locating and editing factual associations in gpt. *Advances in neural information processing systems*, 35:17359–17372.

Zhang, T., Bi, S., Hong, Y., Zhang, K., Luan, F., Yang, S., Sunkavalli, K., Freeman, W. T., and Tan, H. (2025). Test-time training done right. *arXiv preprint arXiv:2505.23884*.