Benchmarking Optimizers:
for Large Language Model Pretraining

Andrei Semenov, Matteo Pagliardini, Martin Jaggi

ML&O EPFL

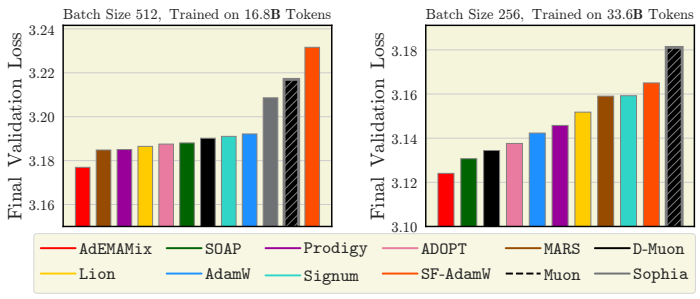
Adam vs. the New Wave of Methods

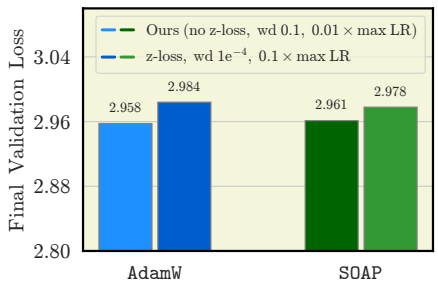
The long-standing baseline:

- **`Adam(W)`** has dominated deep learning for more than a decade
- New optimizers challenge this status-quo: **`Muon`**, **`SOAP`**, **`AdEMAMix`**, **`MARS`**, ...

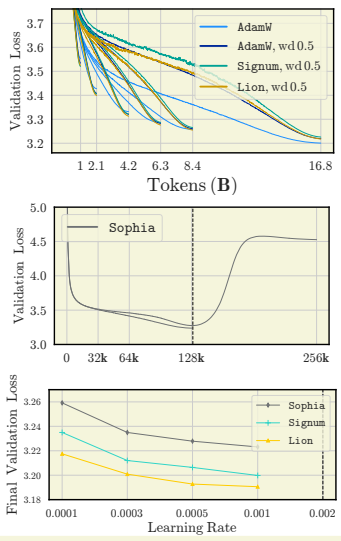
Problems:

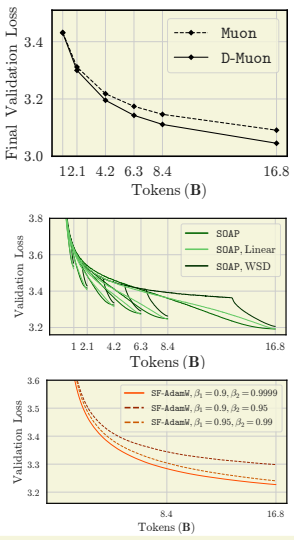
- Need to compare all optimizers in an unified setting
- Rigorous tuning of all methods
- Results vary with the compute budget

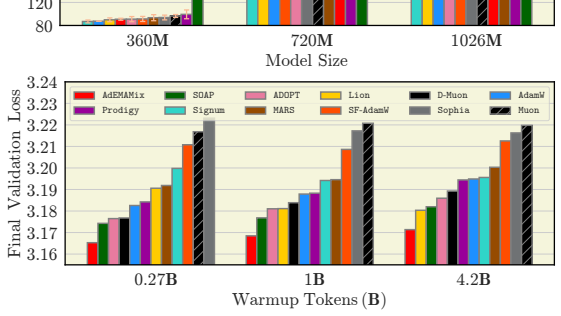




Revisiting Pretraining Best-practices





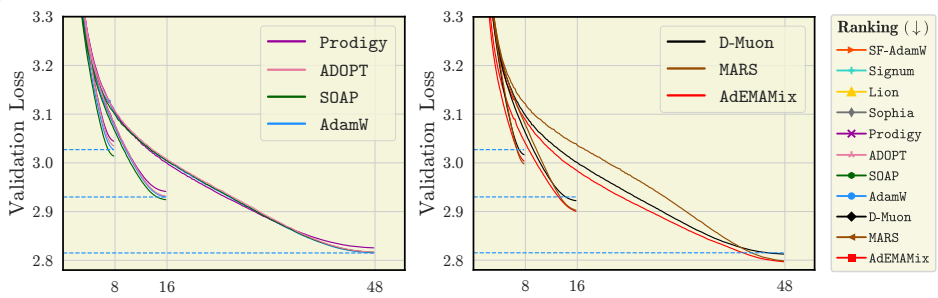


Takeaways & Recommendations

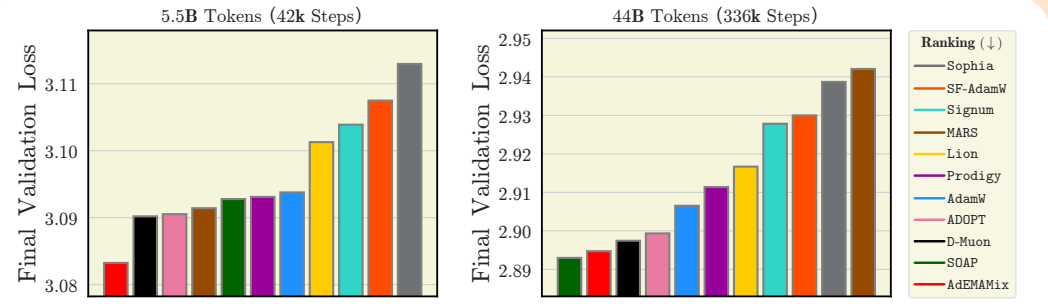
- New optimizers outperform **`Adam`**. **`AdEMAMix`**, **`D-Muon`**, **`SOAP`** are the most consistent
- **`MARS`** is performing on large batches
- **`Prodigy`** performs similar to **`AdamW`** but needs significantly less HP tuning
- Matrix-based optimizers allow using larger LR compared to **`Adam`**-like and sign-based method
- Sign-based methods need smaller LR and longer warmup.
- Dense-to-MoE trends mostly transfer

Benchmarking Results

Dense Models (720M, 1M BS)



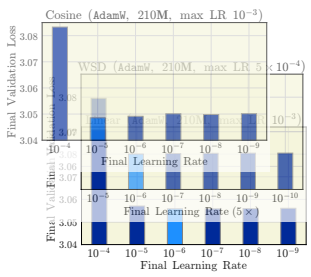
MoEs (520M, 130k BS)

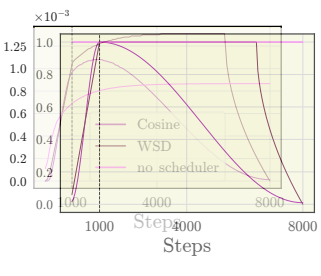


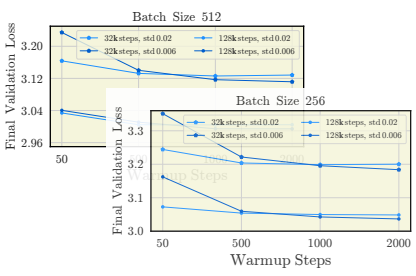
More Ablations...

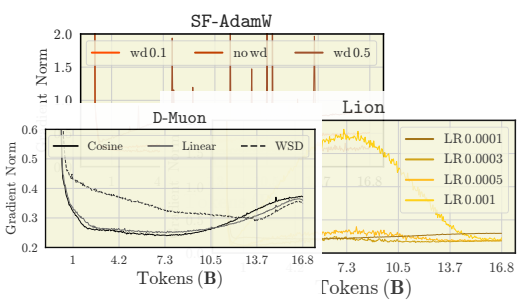
12 optimizers, more than 2.9k runs, 30k GPU hours, and much more experiments

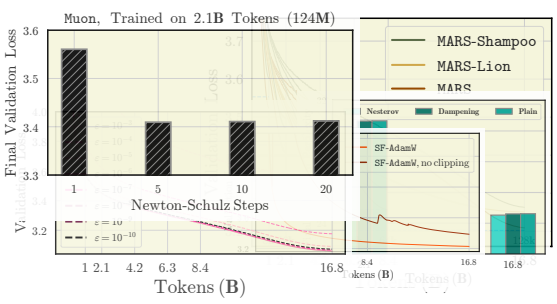
LR Decay **Eff. LR of `Prodigy`** **Weight Init.** **Grad. Norm Patterns** **Optimizer-related Phenomena**











Contact: andrii.semenov@epfl.ch
Code: github.com/epfml/llm-optimizer-benchmark
Paper: arxiv.org/abs/2509.01440
Logs: wandb.ai/semenov-andrei-v/llm-optimizer-benchmark
References: Jordan et al. (2024): *`Muon`*, Kimi (2025): *`D-Muon`*, Vyas et al. (2024): *`SOAP`*, Pagliardini et al. (2024): *`AdEMAMix`*, Yuan et al. (2024): *`MARS`*