# Understanding Optimizer Performance in Language Model Pretraining: A Case Study of Sophia Variants

Aardvark

October 24, 2025

**Abstract**

This paper presents a rigorous empirical evaluation of optimizer performance in language model pretraining, focusing on modifications to the Sophia optimizer. We conduct extensive experiments on the FineWeb dataset using a 134M parameter Transformer model, comparing our SophiaG+ variant against eight existing approaches. While our method combines fast and slow momentum terms with adaptive Hessian scaling, it achieves a validation loss of 5.17, underperforming both AdamW (4.93) and the original Sophia (5.09). Through detailed analysis of training dynamics and parameter sensitivity, we identify key challenges in adapting second-order methods for language model optimization. We provide actionable insights for future research and release complete implementation details to facilitate reproduction.

## 1 Introduction

The optimization landscape for large language models has evolved significantly since the dominance of AdamW [?]. Recent work has introduced second-order methods like Sophia [?] and hybrid approaches like LAMVS [?] and StableAdamW [?]. This paper examines whether Sophia's performance can be improved through careful modifications to its momentum handling, while providing broader insights into optimizer behavior during pretraining.

## 2 Related Work

Modern language model optimization builds upon several key developments. AdamW [?] established weight decay decoupling as crucial for stable training. Subsequent work introduced curvature-aware methods like Sophia [?] and hybrid approaches including:

- LAMVS's layer-wise adaptation [?]

- StableAdamW's variance control [?]
- Adaptive momentum techniques [?]

# 3 Methodology

## 3.1 Base Optimizers

We compare against AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.999$) and Sophia ($\beta_1 = 0.965$, $\beta_2 = 0.99$, $\rho = 0.04$), following their standard formulations.

## 3.2 SophiaG+ Formulation

Our variant modifies Sophia in two ways:

$$m_t^{combined} = 0.7 m_t^{fast} + 0.3 m_t^{slow} \tag{1}$$

$$\rho_t = \rho_0 (1 + \alpha \log(1 + t)) \tag{2}$$

where $m_t^{fast}$ uses $\beta_1 = 0.9$, $m_t^{slow}$ uses $\beta_2 = 0.999$, and $\alpha = 0.1$.

# 4 Experimental Setup

## 4.1 Model and Data

We use a 134M parameter Transformer trained on FineWeb for 640 steps (4M tokens/step). All runs used the same random seed and hyperparameters:

- Learning rate: $10^{-3}$ (linear warmup)
- Weight decay: 0.1
- Batch size: 512 sequences

## 4.2 Evaluation Protocol

We evaluate using: 1. Final validation loss 2. Training curve dynamics 3. Parameter update statistics

# 5 Results

# 6 Analysis

## 6.1 Training Dynamics

The slower convergence of SophiaG+ suggests its momentum combination may interfere with Hessian adaptation. Figure 1 (omitted) shows oscillatory behavior not present in baseline Sophia.

| Optimizer | Validation Loss |
|---|---|
| LAMVS [?] | 4.82 |
| StableAdamW [?] | 4.92 |
| AdamW [?] | 4.93 |
| Stable Momentum [?] | 5.04 |
| Sophia [?] | 5.09 |
| SophiaG+ (Ours) | 5.17 |
| Scaled VR Momentum [?] | 5.26 |
| Adaptive Momentum [?] | 5.34 |
| Subspace-Adaptive [?] | 6.36 |

Table 1: Validation losses across optimizers

### 6.2 Limitations

Key constraints of our study: 1. Single model size (134M parameters) 2. Fixed training duration (640 steps) 3. Limited hyperparameter exploration

## 7 Conclusion

While our SophiaG+ modifications did not improve upon Sophia, this systematic comparison provides valuable insights for optimizer development. Future work should investigate more sophisticated momentum-curvature interactions and conduct larger-scale evaluations.