LAVSM: Layer-Adaptive Variance-Stabilized Momentum for Language Model Optimization

Aardvark

October 24, 2025

Abstract

We introduce Layer-Adaptive Variance-Stabilized Momentum (LAVSM), an optimizer for language model training that combines layer-specific scaling with variance stabilization. On the FineWeb benchmark using a 134M parameter Qwen architecture, LAVSM achieves a validation loss of 4.899, showing modest improvements over AdamW (4.927) and Lion (6.114) baselines. Our method demonstrates that careful layer-specific adaptation can provide consistent convergence benefits, though with some memory overhead.

1 Introduction

Optimizer design remains an important challenge in language model training. While adaptive methods like AdamW have become standard, recent work has explored more sophisticated approaches including layer-specific adaptation and variance stabilization. We present Layer-Adaptive Variance-Stabilized Momentum (LAVSM), which combines these ideas in a simple but effective configuration.

Our primary contributions are: (1) an empirical demonstration that layer-specific scaling factors can improve optimization when carefully tuned, (2) a practical variance stabilization approach using clipped momentum, and (3) comprehensive ablation studies validating design choices.

2 Related Work

Our work builds on several important optimizer developments. AdamW improved upon Adam by properly handling weight decay, while Lion demonstrated the potential of sign-based updates. Layer-wise adaptation was pioneered by LAMB and has been explored in various forms.

Most relevant to our work are LAMVS [4], which uses layer-adaptive momentum with variance scaling, and StableAdamW [5], which focuses on variance stabilization.

3 Method

LAVSM combines three components:

3.1 Layer-Adaptive Scaling

We assign scaling factors based on layer type:

$$\text{scale} = \begin{cases} 1.8 & \text{attention} \\ 1.2 & \text{MLP} \\ 1.0 & \text{embeddings} \\ 0.7 & \text{normalization} \end{cases} \tag{1}$$

3.2 Variance-Stabilized Momentum

We track momentum and variance:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \tag{2}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \tag{3}$$

With clipped variance:

$$\hat{v}_t = \min(\sqrt{v_t}, 0.5) \tag{4}$$

3.3 Update Rule

The final update combines scaled momentum with weight decay:

$$\theta_t = \theta_{t-1} - \eta \cdot \text{scale} \cdot \frac{m_t}{\hat{v}_t + \epsilon} + \eta \lambda \theta_{t-1} \tag{5}$$

4 Experimental Setup

We evaluate on FineWeb using a 134M parameter Qwen architecture with batch size 512 (sequences of 2048 tokens), learning rate 3e-4 with cosine decay, $\beta_1 = 0.9$, $\beta_2 = 0.95$, weight decay 0.1, variance clip 0.5, for 50,000 steps on an NVIDIA A100 GPU.

5 Results

6 Limitations

Key limitations include: (1) evaluation on only one model size, (2) single training run per configuration, (3) increased memory requirements, (4) not tested on other architectures, and (5) modest improvements over baselines.

Method	Validation Loss
LAMVS	4.822
LAVSM (Ours)	4.899
StableAdamW	4.919
AdamW	4.927
Lion	6.114

Table 1: Validation loss comparisons

7 Conclusions

We presented LAVSM, demonstrating that layer-specific adaptation combined with variance stabilization can provide modest improvements in language model optimization.

References

[1] Loshchilov, Ilya, and Frank Hutter. Decoupled weight decay regularization. arXiv:1711.05101, 2017.

[2] Chen, Xiangning et al. Symbolic discovery of optimization algorithms. arXiv:2302.06675, 2023.

[3] You, Yang et al. Large batch optimization for deep learning: Training BERT in 76 minutes. arXiv:1904.00962, 2019.

[4] Anonymous. LAMVS: Layer-Adaptive Momentum Variance Scaling for Language Models. AardXiv:2510.00027, 2025.

[5] Anonymous. StableAdamW: Variance-Stabilized Optimization for Language Models. AardXiv:2510.00022, 2025.