AdaShift: Decorrelation and Convergence of Adaptive Learning Rate Methods

Zhiming Zhou*, Qingru Zhang*, Guansong Lu, Hongwei Wang, Weinan Zhang, Yong Yu
Apex Data & Knowledge Management Lab, Shanghai Jiao Tong University

上海交通大学 SHANGHAI JIAO TONG UNIVERSITY

Background

Adaptive learning rate methods

- General updating rule: $\theta_{t+1} = \theta_t - \frac{\alpha_t}{\sqrt{v_t}} m_t$.
- Adam:

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t$$
$$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2$$

Non-convergence of Adam

- Reddi et al. (2018) argued that the issue lies in quantity $\Gamma_t \triangleq \left(\frac{\sqrt{v_t}}{\alpha_t} - \frac{\sqrt{v_{t-1}}}{\alpha_{t-1}}\right)$.
- AMSGrad keeps v_t non-decreasing to address the issue, but slows down the training.

Our Contributions

The analysis

- Large gradients tend to have relatively small step sizes and vice versa.
- It is due to the inappropriate positive correlation between v_t and g_t.

The proposed solution:

- Decorrelating v_t and g_t.
- Calculating v_t using temporal shifted g_t.

The Counterexamples

Sequential version:

$$f_t(\theta) = \begin{cases} C\theta, & if\ t\ mod\ d = 1; \\ -\theta, & otherwise. \end{cases}$$

Stochastic version:

$$f_t(\theta) = \begin{cases} C\theta, & with\ probability\ p = \frac{1+\sigma}{C+1}; \\ -\theta, & with\ probability\ p = \frac{C-\sigma}{C+1}. \end{cases}$$

Theoretical Analysis

The proposed analysis tool: net update factor

$$net(g_t) \overset{\text{def}}{=} \sum_{i=t}^{\infty} \frac{\alpha_i}{\sqrt{v_i}} [(1-\beta_1)\beta_1^{i-t} g_t] = k(g_t) g_t$$

where $k(g_t) = \sum_{i=t}^{\infty} \frac{\alpha_i}{\sqrt{v_i}} [(1-\beta_1)\beta_1^{i-t}]$

- SGD and Moment method:

$$k(g_t) = 1$$

- Adam on these counterexamples:

$$k(C) < k(-1)$$

- When v_t and g_t are decorrelated (independent):

$\mathbb{E}[k(g_t)]$ is identical for each g_t

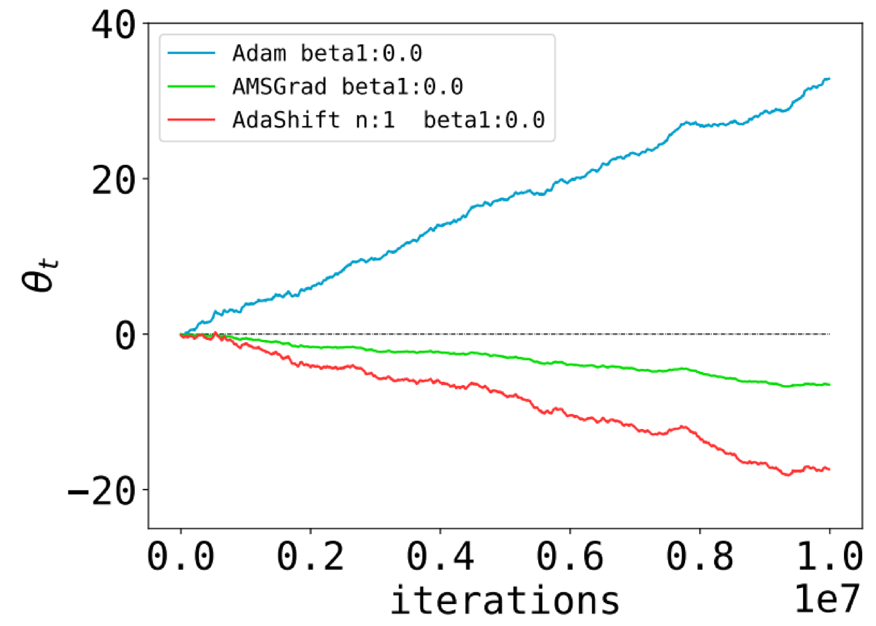
Algorithm: AdaShift

Input: n, β_1, β_2, ϕ, θ_0, $\{f_t(\theta)\}_{t=1}^T$, $\{\alpha_t\}_{t=1}^T$, $\{g_{-t}\}_{t=0}^{n-1}$,

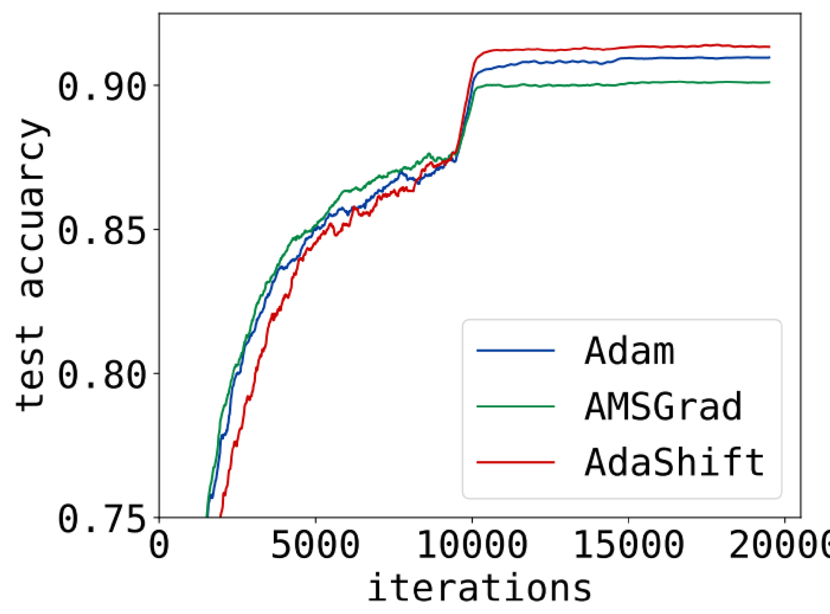
1: set $v_0 = 0$
2: **for** $t = 1$ **to** T **do**
3: $\quad g_t = \nabla f_t(\theta_t)$
4: $\quad m_t = \sum_{i=0}^{n-1} \beta_1^i g_{t-i} / \sum_{i=0}^{n-1} \beta_1^i$
5: $\quad$ **for** $i = 1$ **to** M **do**
6: $\quad\quad v_t[i] = \beta_2 v_{t-1}[i] + (1-\beta_2)\phi(g_{t-n}^2[i])$
7: $\quad\quad \theta_t[i] = \theta_{t-1}[i] - \alpha_t / \sqrt{v_t[i]} \cdot m_t[i]$
8: $\quad$ **end for**
9: **end for**

Experiments

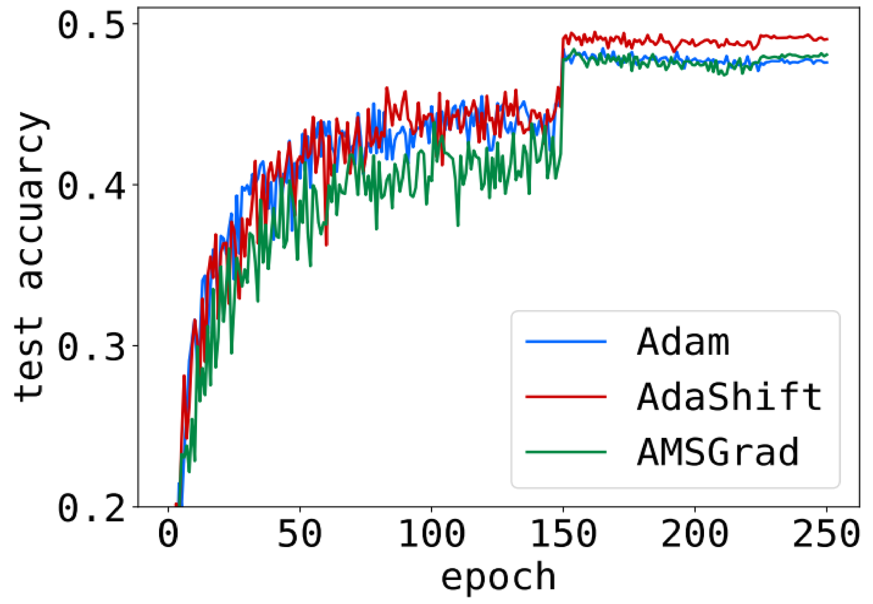
Counterexamples



Resnet on Cifar10



DenseNet on Tiny-ImageNet



Neural Machine Translation BELU