Surge Phenomenon in Optimal Learning Rate and Batch Size Scaling

Shuaipeng Li[1], Penghao Zhao[1,2], Hailin Zhang[1,2], Xingwu Sun[1,3], Hao Wu[1], Dian Jiao[1], Weiyan Wang[1], Chengjun Liu[1], Zheng Fang[1], Jinbao Xue[1], Yangyu Tao[1], Bin Cui[2], Di Wang[1]

[1] Tencent, [2] Peking University, [3] University of Macau

NEURAL INFORMATION PROCESSING SYSTEMS

Tencent 腾讯

北京大學 PEKING UNIVERSITY

澳門大學 UNIVERSIDADE DE MACAU UNIVERSITY OF MACAU

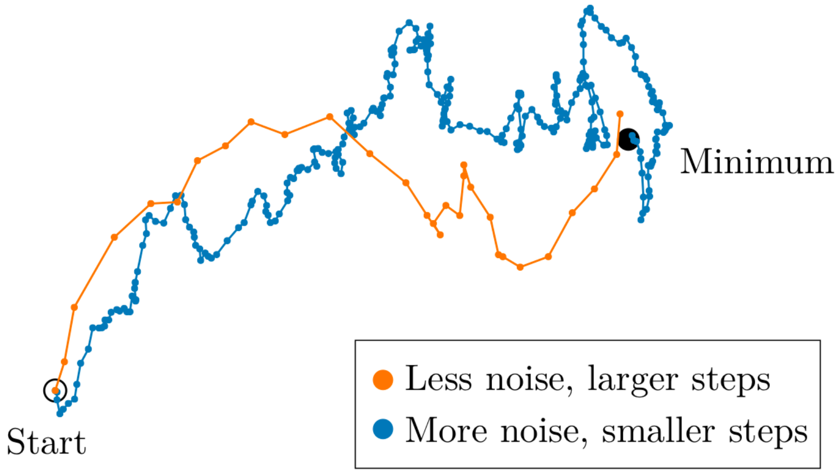
Optimal Learning Rate: Related to Batch Size

- Past experience: **larger** batches require **larger** optimal learning rates.

- Intuition from **gradient noise**:
  - Small batch size -> high noise -> small step size.
  - Large batch size -> low noise -> large step size.

- Current insights into scaling:
  - **Linear** or **Sqrt**.

- Significance:
  - Guide model training, **simplify hyper-parameter tuning**.



OpenAI, An Empirical Model of Large-Batch Training, 2018

Research on SGD Optimizer [OpenAI, 2018]

- Perturb the parameter and apply a Taylor expansion:

$$L(\theta - \epsilon V) \approx L(\theta) - \epsilon G^T V + \frac{1}{2}\epsilon^2 V^T H V.$$

$$\mathbb{E}_{x_{1\cdots B}\sim\rho}\left[G_{\text{est}}(\theta)\right] = G(\theta)$$

$$\text{cov}_{x_{1\cdots B}\sim\rho}\left(G_{\text{est}}(\theta)\right) = \frac{1}{B}\Sigma(\theta),$$

$$\Sigma(\theta) \equiv \text{cov}_{x\sim\rho}\left(\nabla_\theta L_x(\theta)\right) = \mathbb{E}_{x\sim\rho}\left[\left(\nabla_\theta L_x(\theta)\right)\left(\nabla_\theta L_x(\theta)\right)^T\right] - G(\theta)\,G(\theta)^T.$$

- G_{est} contains noises. Calculate expectation:

$$\mathbb{E}[L(\theta - \epsilon G_{\text{est}})] = L(\theta) - \epsilon|G|^2 + \frac{1}{2}\epsilon^2\left(G^T H G + \frac{\text{tr}(H\Sigma)}{B}\right).$$

- Minimizing the equation:

$$\epsilon_{\text{opt}}(B) = \text{argmin}_\epsilon \mathbb{E}\left[L(\theta - \epsilon G_{\text{est}})\right] = \frac{\epsilon_{\max}}{1 + \mathcal{B}_{\text{noise}}/B}$$

$$\epsilon_{\max} \equiv \frac{|G|^2}{G^T H G}$$

$$\Delta L_{\text{opt}}(B) = \frac{\Delta L_{\max}}{1 + \mathcal{B}_{\text{noise}}/B}; \qquad \Delta L_{\max} = \frac{1}{2}\frac{|G|^4}{G^T H G}.$$

$$\mathcal{B}_{\text{noise}} = \frac{\text{tr}(H\Sigma)}{G^T H G},$$

- Approximations：
  - When B << B_{noise}, the optimal learning rate scales **linearly** with the batch size.
  - When B >> B_{noise}, the optimal learning rate **approaches its maximum value**.

Trading-off # training steps and # samples [OpenAI, 2018]

- # training steps: $\delta S = 1 + \frac{\mathcal{B}}{B}$; # samples: $\delta E = B\delta S$
- Considering multiple steps:

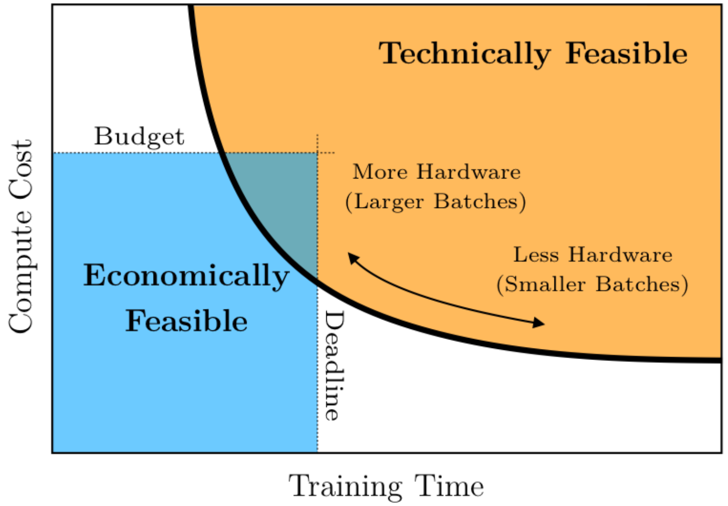
$$S = \int \left(1 + \frac{\mathcal{B}(s)}{B(s)}\right) ds$$

$$E = \int (\mathcal{B}(s) + B(s))\, ds$$

- After some derivation:

$$\frac{S}{S_{\min}} - 1 = \left(\frac{E}{E_{\min}} - 1\right)^{-1}$$

- **Need to balance the computation cost and training time!**



Adam optimizer is different from SGD optimizer

- The main difference is the **update amount** at each step:
  - Simplified assumption: the update amount is the **gradient's sign**:

$$\theta_{i+1} = \theta_i - \epsilon \cdot sign(G_{est}),$$

  - In Adam, the update amount is:

$$V = \frac{\widehat{m}_t}{\sqrt{\widehat{v}_t} + \epsilon_{Adam}} = \frac{\frac{1-\beta_1}{1-\beta_1^t}\sum_i^t \beta_1^{t-i} G_{est,i}}{\sqrt{\frac{1-\beta_2}{1-\beta_2^t}\sum_i^t \beta_2^{t-i} G_{est,i}^2} + \epsilon_{Adam}}$$

  - β->1: the following formula tends to sign(G_{est}) when the variance is small

$$V = \frac{\frac{\sum_i^t G_{est,i}}{t}}{\sqrt{\frac{\sum_i^t G_{est,i}^2}{t}}} = \frac{\mathbb{E}_t[G_{est}]}{\sqrt{\mathbb{E}_t[G_{est}^2]}} = \frac{sign(\mathbb{E}_t[G_{est}])}{\sqrt{1 + \frac{var_t(G_{est})}{\mathbb{E}_t[G_{est}]^2}}}$$

  - β->0: degenerates to sign(G_{est})

$$V = \frac{G_{est}}{\sqrt{G_{est}^2}} = sign(G_{est})$$

"Surge Phenomenon in Optimal Learning Rate and Batch Size Scaling." NeurIPS 2024

The optimal learning rate for Adam optimizer

- Still Taylor expansion:

$$\Delta L_{opt} = \frac{G^T \mathbb{E}[V]}{2} \epsilon_{opt}.$$

$$\mathbb{E}[\Delta L] = \mathbb{E}[L(\theta) - L(\theta - \epsilon \cdot V)] \approx \epsilon G^T \mathbb{E}[V] - \frac{1}{2}\epsilon^2 \mathbb{E}[V^T H V]. \Rightarrow \epsilon_{opt} \equiv argmax_{\epsilon}\, \mathbb{E}[\Delta L] = \frac{G^T \mathbb{E}[V]}{tr[H \cdot cov(V)] + \mathbb{E}[V]^T H \mathbb{E}[V]},$$

- For the update amount sign(G_{est}):
  - Assumption: the gradient follows a **normal distribution**.
  - For each sample, the expectation is $erf(\frac{\mu}{\sqrt{2}\sigma})$ and the variance is $1 - erf(\frac{\mu}{\sqrt{2}\sigma})^2$
  - For a batch, we have $G_{est}(\theta_i) = \frac{1}{B}\sum^{B} G_x(\theta_i) \sim \mathcal{N}(\mu_i, \frac{\sigma_i^2}{B})$
  - For all parameters: $\mathbb{E}[V] = \begin{pmatrix} \vdots \\ erf(\sqrt{\frac{B}{2}}\frac{\mu_i}{\sigma_i}) \\ \vdots \end{pmatrix} \quad cov(V) = \begin{pmatrix} \ddots & & 0 \\ & 1 - erf(\sqrt{\frac{B}{2}}\frac{\mu_i}{\sigma_i})^2 & \\ 0 & & \ddots \end{pmatrix}$
- As a result:

$$\Delta L_{opt} = \frac{1}{2}\frac{\sum_i \sum_j \mathcal{E}_i \mathcal{E}_j \mu_i \mu_j}{\sum_i (1 - \mathcal{E}_i^2) H_{i,i} + \sum_i \sum_j \mathcal{E}_i \mathcal{E}_j H_{i,j}} \quad \epsilon_{opt} = \frac{\sum_i \mathcal{E}_i \mu_i}{\sum_i (1 - \mathcal{E}_i^2) H_{i,i} + \sum_i \sum_j \mathcal{E}_i \mathcal{E}_j H_{i,j}} \Leftarrow \mathcal{E}_i(B) = \frac{2}{\sqrt{\pi}} \int_0^{\sqrt{\frac{B}{2}}\frac{\mu_i}{\sigma_i}} e^{-t^2} dt \approx \frac{\frac{\mu_i}{\sigma_i}}{\sqrt{\frac{\pi}{2B} + (\frac{\mu_i}{\sigma_i})^2}}$$

The optimal learning rate for Adam optimizer

- Simplifying the relationship between learning rate and batch size:

$$\epsilon(B) = \frac{\beta f(B)}{f(B)^2 + \gamma} = \frac{\beta}{f(B) + \frac{\gamma}{f(B)}}$$

- When $B \ll \frac{\pi\sigma_i^2}{2\mu_i^2}$, the optimal learning rate **first rises then drops**:

$$\epsilon_{opt}(B) \approx \frac{1}{\frac{1}{2}(\sqrt{\frac{\mathcal{B}_{noise}}{B}} + \sqrt{\frac{B}{\mathcal{B}_{noise}}})} \frac{\sqrt{\frac{\mathcal{B}_{noise}}{2\pi}} \sum_i \frac{\mu_i^2}{\sigma_i}}{\sum_i H_{i,i}} \leq \frac{\sqrt{\frac{\mathcal{B}_{noise}}{2\pi}} \sum_i \frac{\mu_i^2}{\sigma_i}}{\sum_i H_{i,i}} \Leftarrow \mathcal{E}_i(B) \approx \sqrt{\frac{2B}{\pi}} \frac{\mu_i}{\sigma_i} \quad \mathcal{B}_{noise} = \frac{\pi \sum_i H_{i,i}}{2 \sum_i \sum_j \begin{cases} \frac{\mu_i \mu_j}{\sigma_i \sigma_j} & i \neq j \\ 0 & i = j \end{cases} H_{i,j}}$$

- When $B \gg \frac{\pi\sigma_i^2}{2\mu_i^2}$, the optimal learning rate **tends to a constant**:

$$\epsilon_{opt} = \frac{\sum_i |\mu_i|}{\sum_i \sum_j sign(\mu_i) sign(\mu_j) H_{i,j}} \quad \Leftarrow \quad \mathcal{E}_i = sign(\frac{\mu_i}{\sigma_i}) = sign(\mu_i)$$

“Surge Phenomenon in Optimal Learning Rate and Batch Size Scaling.” NeurIPS 2024

Trading-off # training steps and # samples with Adam optimizer

- The **same trade-off** is also obtained with the Adam optimizer:

$$\left(\frac{S}{S_{min}}-1\right)\left(\frac{E}{E_{min}}-1\right)=1.$$

Adam **only affects ΔL_{max}**, does not change the equation!

- When $B \ll \frac{\pi\sigma_i^2}{2\mu_i^2}$, the loss delta is:

$$\Delta L_{opt}(B)=\frac{\Delta L_{max}}{1+\frac{\mathcal{B}_{noise}}{B}} \Leftarrow \Delta L_{max}=\frac{\sum_i\sum_j \frac{\mu_i^2\mu_j^2}{\sigma_i\sigma_j}}{2\sum_i\sum_j \begin{cases}\frac{\mu_i\mu_j}{\sigma_i\sigma_j} & i\neq j\\ 0 & i=j\end{cases} H_{i,j}}$$

- This formula is the same as the SGD case! Consequently,

$$S=\int\left(1+\frac{\mathcal{B}_{noise}}{B}\right)ds \qquad E=\int(\mathcal{B}_{noise}+B)ds \Rightarrow \left(\frac{S}{S_{min}}-1\right)\left(\frac{E}{E_{min}}-1\right)=1.$$

"Surge Phenomenon in Optimal Learning Rate and Batch Size Scaling." NeurIPS 2024

Application process

- In the experiment, we **fit B_{noise}** with the following formula:

$$\frac{1}{S} = -\mathcal{B}_{noise}\frac{1}{E} + \frac{1}{S_{min}}$$

$$B_{peak} = \mathcal{B}_{noise} \approx \mathcal{B}_{crit} = \frac{E_{min}}{S_{min}} \approx \frac{B_*}{L^{\frac{1}{\alpha_B}}}$$

$$B_{\text{crit}}(L) = \frac{B_*}{L^{1/\alpha_B}}, \qquad B_* \sim 2 \cdot 10^8 \text{ tokens}, \quad \alpha_B \sim 0.21$$

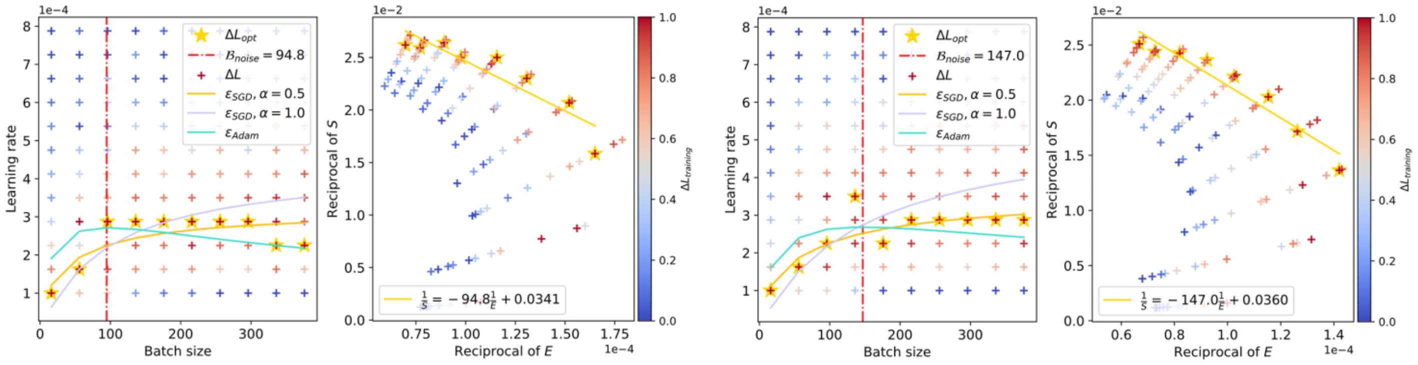
- In practice, we can also derive it through the scaling law above.
- And then use one pair of (optimal learning rate, batch size) to **obtain ε_{max}**:

$$\mathbb{E}[\epsilon_{max}]_{Adam} = \mathbb{E}[\frac{\epsilon_{opt}}{2}(\sqrt{\frac{\mathcal{B}_{noise}}{B}} + \sqrt{\frac{B}{\mathcal{B}_{noise}}})]$$

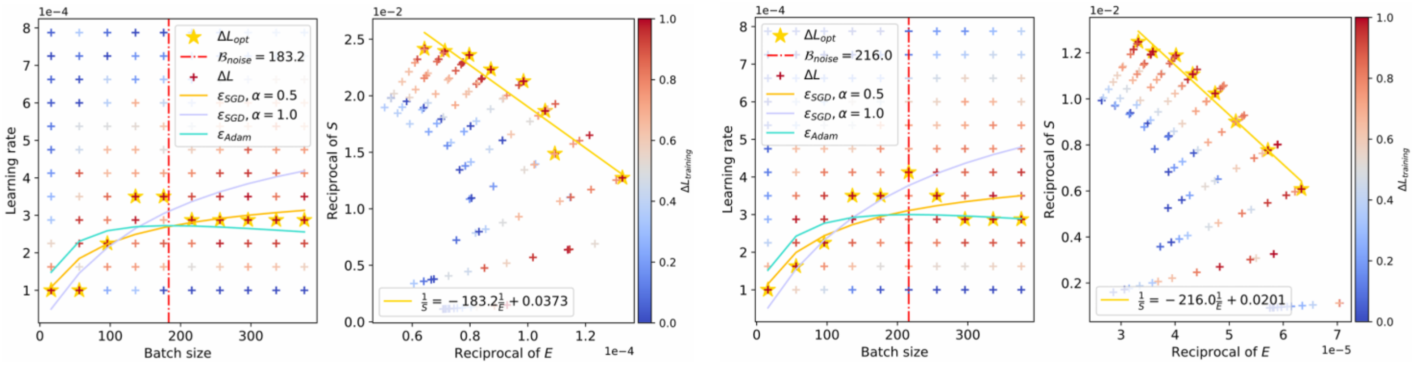
$$\mathbb{E}[\epsilon_{max}]_{SGD} = \mathbb{E}[\epsilon_{opt}(1 + \frac{\mathcal{B}_{noise}}{B})^{\alpha}]$$

Experiments: multiple workloads including CV and NLP

- ResNet-18 on TinyImageNet



(a) $L_{training} = 5.213 \pm 0.001$ (b) $L_{training} = 5.146 \pm 0.001$

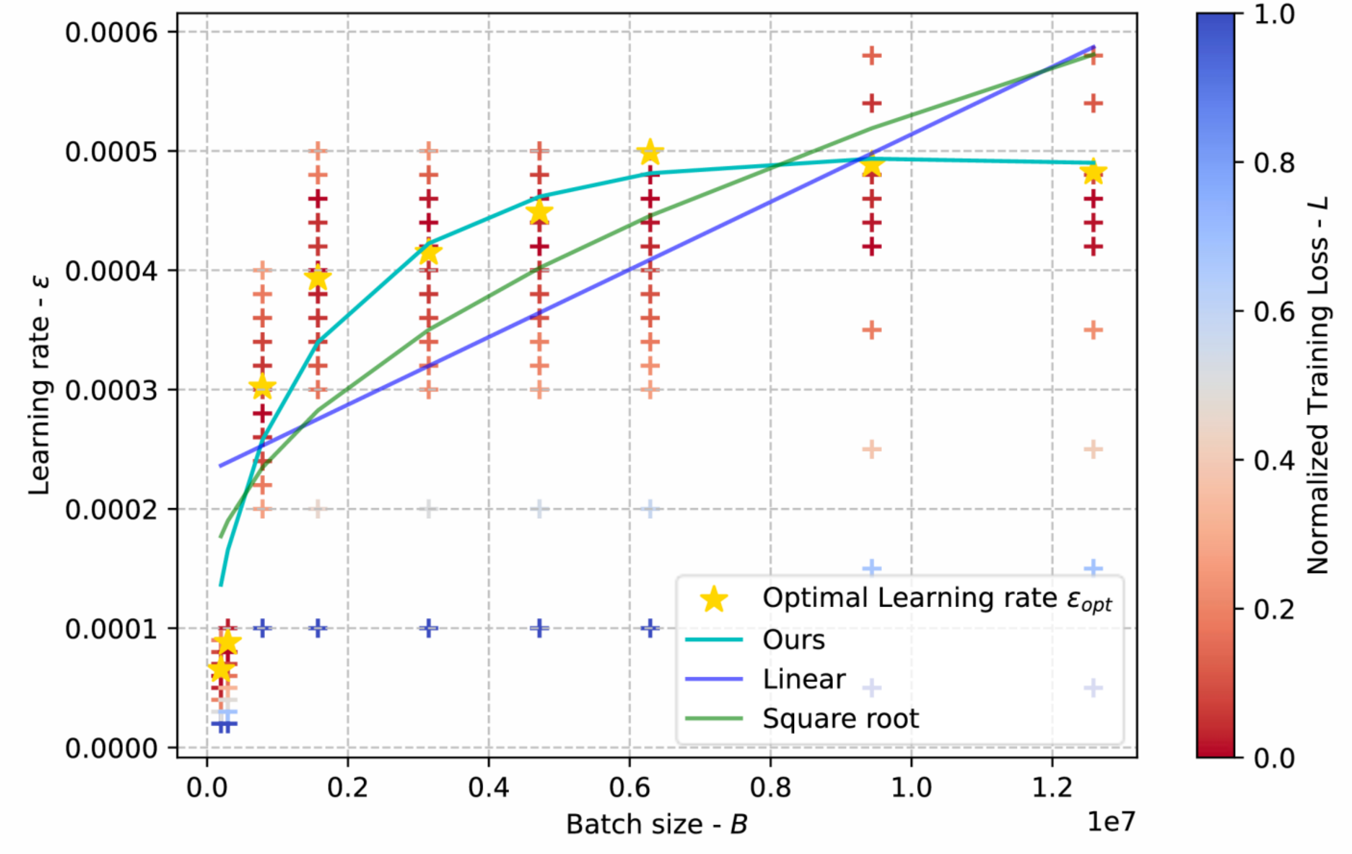


(c) $L_{training} = 5.113 \pm 0.001$ (d) $L_{training} = 4.863 \pm 0.001$

"Surge Phenomenon in Optimal Learning Rate and Batch Size Scaling." NeurIPS 2024

Experiments: multiple workloads including CV and NLP

- MoE on RedPajama-v2



"Surge Phenomenon in Optimal Learning Rate and Batch Size Scaling." NeurIPS 2024

Conclusion and significance

- Takeaways:
  - As the batch size increases, the optimal learning rate demonstrates a **decreasing trend** within a specified range.
  - The batch size that corresponds to the local maximum optimal learning rate **is consistent with** the balance point of training speed and data efficiency. As the training progresses and the loss decreases, B_{noise} will gradually **becomes larger**.

- Significance:
  - A deeper understanding of the **training dynamics**.
  - Help **tune hyperparameters**, improve convergence speed, and avoid complicated grid searches.

Thanks for your attention!