Sharpness-Aware Training for Free

Jiawei Du[1,2], Daquan Zhou [3], Jiashi Feng[3], Vincent Y. F. Tan [4,2] Joey Zhou Tianyi[1,2]

[1] Centre for Frontier AI Research (CFAR), A*STAR, Singapore [2] Dept. ECE, National University of Singapore, Singapore
[3] ByteDance Inc [4] Dept. Mathmatics, National University of Singapore, Singapore

Introduction

- Many research works [1,2] argue that the geometry of a Deep Neural Network's (DNN's) loss landscape affects generalization and DNNs with flat minima can generalize better.
- Sharpness-Aware Training [3,4] helps DNNs to converge to a flat region by regularizing a sharpness measure. However, the calculation of the sharpness measure results in computational overhead being doubled (2X).

Contributions

- We propose a novel trajectory loss to measure the sharpness to be used for sharpness-aware training, which requires almost zero extra computational overhead. This is the the sharpness-aware training for free (or SAF) algorithm.
- We propose SAF and a memory-efficient variant MESA based on the trajectory loss to improve DNN's generalization ability.

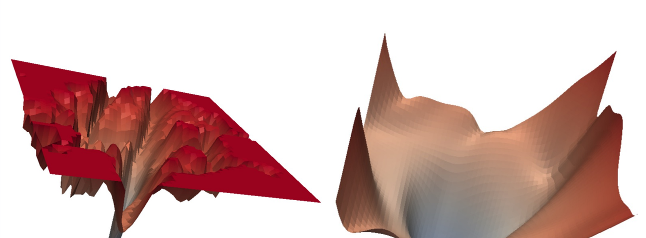


Figure 1: Loss landscapes of a sharp minimum and a flat minimum.

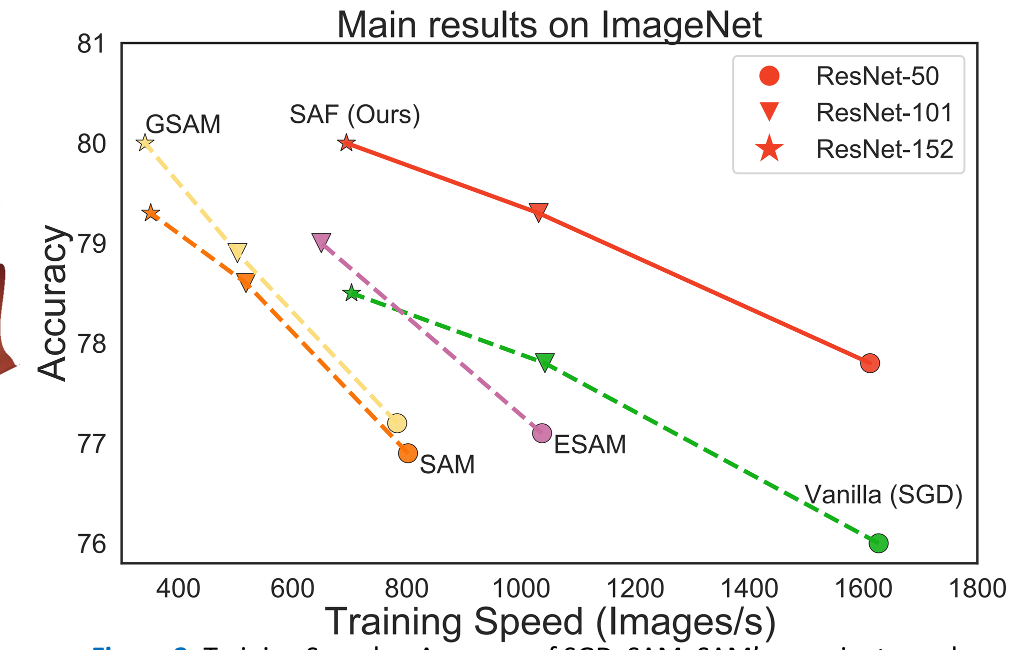


Figure 2: Training Speed vs Accuracy of SGD, SAM, SAM's variants, and our proposed SAF.

Preliminaries

Sharpness-Aware Minimization [3]

- **Objective:** trains DNN by solving a minimax optimization problem.

$$\min_{\theta} \underbrace{\boxed{\max_{\epsilon:\|\epsilon\|_2 \le \rho} L_{\mathbb{S}}(f_{\theta+\epsilon}) - L_{\mathbb{S}}(f_{\theta})}}_{\text{Sharpness Measure}} + L_{\mathbb{S}}(f_\theta) + \lambda\|\theta\|_2^2$$

where $\mathbb{S} \triangleq \{(x_i, y_i)\}_{i=1}^n$ is drawn i.i.d. from a natural distribution $\mathcal{D}$
f_θ: neural networks with weights θ; L: loss function
ϵ: weight perturbation; ρ, λ: given hyperparameters

The Sharpness Measure is defined as

$$R_{\mathbb{S}}(f_\theta) = \max_{\epsilon:\|\epsilon\|_2 \le \rho} L_{\mathbb{S}}(f_{\theta+\epsilon}) - L_{\mathbb{S}}(f_\theta) = L_{\mathbb{S}}(f_{\theta+\hat{\epsilon}}) - L_{\mathbb{S}}(f_\theta)$$

where $\hat{\epsilon} = \arg\max_{\epsilon:\|\epsilon\|_2<\rho} L_{\mathbb{S}}(f_{\theta+\epsilon}) \approx \rho \frac{\nabla_\theta L_{\mathbb{S}}(f_\theta)}{\|\nabla_\theta L_{\mathbb{S}}(f_\theta)\|}$

Objective: To find a "cheaper" replacement of the sharpness measure.

Full Paper is Available at:

https://arxiv.org/abs/2205.14083

Method: Leverage the trajectory of weights to estimate the sharpness

Objective: Find a "cheaper" replacement of the sharpness measure. (SAF)
In t^{th} iteration, a mini-batch $\mathbb{B}_t \subset \mathbb{S}$ is sampled for optimizing θ_t,
We define $\gamma_i = \frac{\eta_i}{\rho^2}\cos(\Phi_i)$, $\cos\Phi_i = \frac{\nabla_{\theta_i} L_{\mathbb{B}_t}(f_{\theta_i})^\top \nabla_{\theta_i} L_{\mathbb{B}_i}(f_{\theta_i})}{\|\nabla_{\theta_i} L_{\mathbb{B}_t}(f_{\theta_i})\| \, \|\nabla_{\theta_i} L_{\mathbb{B}_i}(f_{\theta_i})\|}$ we have

$$\begin{aligned}
\arg\min_{\theta_t} R_{\mathbb{B}_t}(f_{\theta_t}) &= \arg\min_{\theta_t} \gamma_t R_{\mathbb{B}_t}(f_{\theta_t}) R_{\mathbb{B}_t}(f_{\theta_t}) \\
&= \arg\min_{\theta_t} [\gamma_t R_{\mathbb{B}_t}(f_{\theta_t}) R_{\mathbb{B}_t}(f_{\theta_t}) + \gamma_i R_{\mathbb{B}_t}(f_{\theta_i}) R_{\mathbb{B}_i}(f_{\theta_i})] \\
&= \mathop{\mathbb{E}}_{\theta_i \sim \mathrm{Unif}(\Theta)} [\gamma_i R_{\mathbb{B}_t}(f_{\theta_i}) R_{\mathbb{B}_i}(f_{\theta_i})]
\end{aligned}$$

where $\Theta = \{\theta_2, \theta_3, \ldots, \theta_{t-1}\}$ is the past trajectory of the weights
Then

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\theta_i \sim \mathrm{Unif}(\Theta)} [\gamma_i R_{\mathbb{B}_t}(f_{\theta_i}) R_{\mathbb{B}_i}(f_{\theta_i})] \\
&\approx \mathop{\mathbb{E}}_{\theta_i \sim \mathrm{Unif}(\Theta)} [\eta_i \cos(\Phi_i) \|\nabla_{\theta_i} L_{\mathbb{B}_t}(f_{\theta_i})\| \, \|\nabla_{\theta_i} L_{\mathbb{B}_i}(f_{\theta_i})\|] \\
&= \mathop{\mathbb{E}}_{\theta_i \sim \mathrm{Unif}(\Theta)} [\eta_i \nabla_{\theta_i} L_{\mathbb{B}_t}(f_{\theta_i})^\top \nabla_{\theta_i} L_{\mathbb{B}_i}(f_{\theta_i})] \\
&\approx \mathop{\mathbb{E}}_{\theta_i \sim \mathrm{Unif}(\Theta)} [L_{\mathbb{B}_t}(f_{\theta_i}) - L_{\mathbb{B}_t}(f_{\theta_{i+1}})] \\
&= \frac{1}{t-1}[L_{\mathbb{B}_t}(f_{\theta_1}) - L_{\mathbb{B}_t}(f_{\theta_t})],
\end{aligned}$$

- Now, we have a **good replacement** of the sharpness measure **without additional computations**.
- However, the loss difference $L_{\mathbb{B}_t}(f_{\theta_1}) - L_{\mathbb{B}_t}(f_{\theta_t})$, because it will cancel out with the vanilla loss $L_{\mathbb{B}_t}(f_{\theta_t})$. Hence, we use KL divergence Loss.

We use a trajectory loss defined below to replace the sharpness measure by using a trajectory loss, thus out method is called sharpness-aware training for free (SAF).

$$L_{\mathbb{B}}^{\text{tra}}(f_\theta, \mathbb{Y}^{(e-\tilde{E})}) = \frac{\lambda}{|\mathbb{B}|} \sum_{x_i \in \mathbb{B}, \hat{y}_i^{(e-\tilde{E})} \in \mathbb{Y}^{(e-\tilde{E})}} \mathrm{KL}\left(\frac{1}{\tau}\hat{y}_i^{(e-\tilde{E})}, \frac{1}{\tau} f_\theta(x_i)\right)$$

where $\mathbb{Y}^{(e-\tilde{E})} = \{\hat{y}_i^{(e-\tilde{E})} = f_\theta^{(e-\tilde{E})}(x_i) : x_i \in \mathbb{B}\}$ e is the current epoch,
$\hat{y}_i^{(e-\tilde{E})}$ is the output of the network (soft logits) of instance x_i in $\tilde{E}$ epochs ago.

- Intuitively, SAF prevents the training from converging to a sharp local minimum by avoiding a sudden drop in the loss during training.

Objective: A memory-efficient version of SAF (MESA).
Motivation of MESA:

1. SAF needs to record/store the outputs of each instances, which incurs an **out-of-memory** issue on very **large-scale** datasets (ImageNet and larger).
2. The most recent iteration's sharpness estimated by SAF will **decay with the learning rate** of the base optimizer.

We adopt an exponential moving average (EMA) model to construct the trajectory loss, which is

$$L_{\mathbb{B}}^{\text{tra}}(f_\theta, f_{v_t}) = \frac{1}{|\mathbb{B}|} \sum_{x_i \in \mathbb{B}} \mathrm{KL}\left(\frac{1}{\tau} f_{v_t}(x_i), \frac{1}{\tau} f_\theta(x_i)\right)$$

v_t is the weights of EMA model, whose outputs are treated as the reference of the trajectory loss.

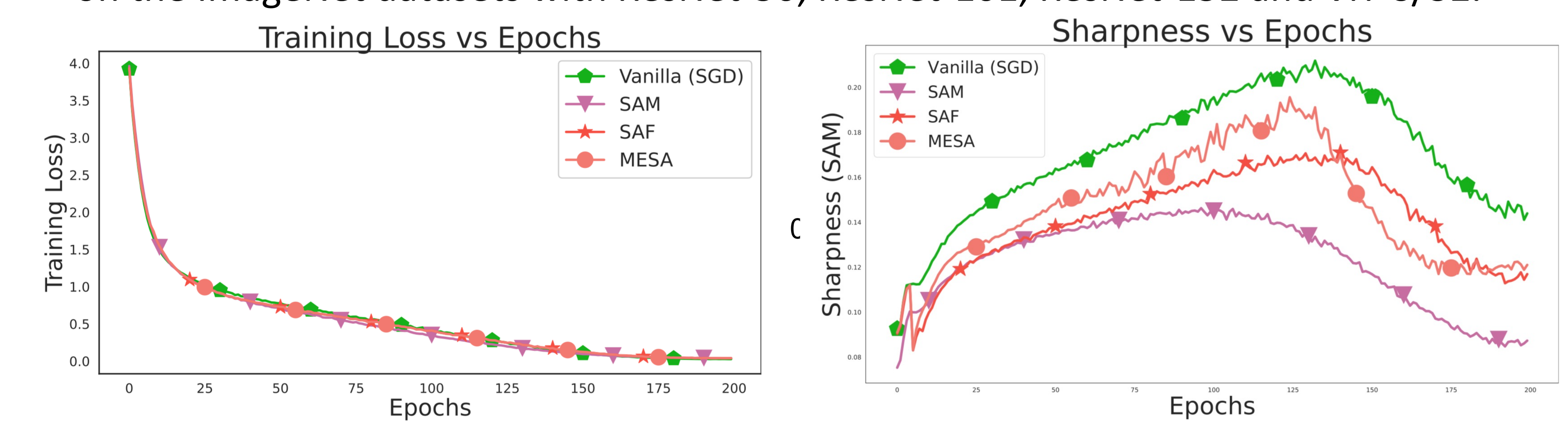
- MESA employs the EMA model to conduct one **forward only** inference (15% additional computations) to save memory.

Experiments:

	ResNet-50		ResNet-101	
ImageNet	Accuracy	images/s	Accuracy	images/s
Vanilla (SGD)	76.0	1,627 (100%)	77.8	1,042 (100%)
SAM [8]	76.9	802 (49.3%)	78.6	518 (49.7%)
ESAM [1] [6]	77.1	1,037 (63.7%)	79.1	650 (62.4%)
GSAM [2] [36]	77.2	783 (48.1%)	78.9	503 (48.3%)
SAF (Ours)	**77.8**	1,612 (99.1%)	**79.3**	1,031 (99.0%)
MESA (Ours)	77.5	1,386 (85.2%)	79.1	888 (85.4%)

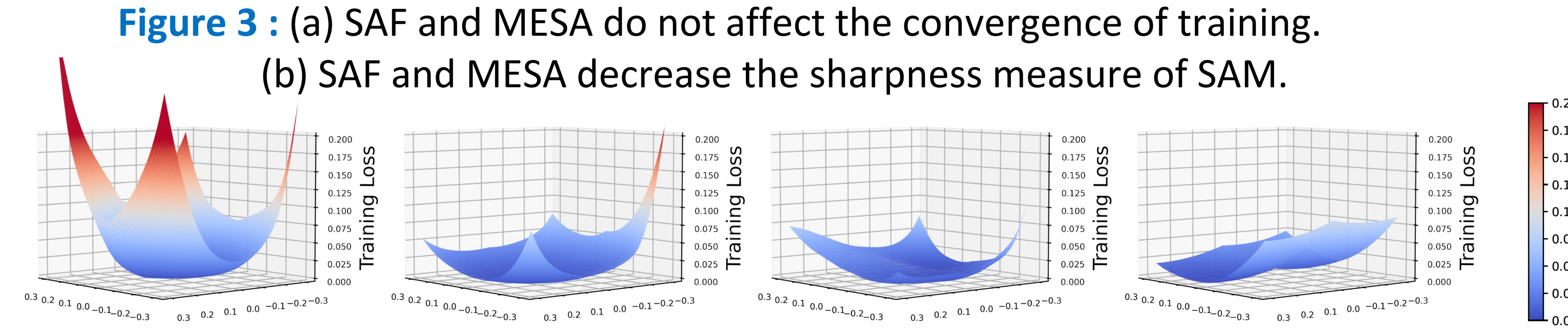
	ResNet-152		ViT-S/32	
ImageNet	Accuracy	images/s	Accuracy	images/s
Vanilla[3]	78.5	703 (100%)	68.1	5,154 (100%)
SAM [8]	79.3	351 (49.9%)	68.9	2,566 (49.8%)
LookSAM[4] [20]	-	-	68.8	4,273 (82.9%)
GSAM[2] [36]	**80.0**	341 (48.5%)	**73.8**	2,469 (47.9%)
SAF (Ours)	79.9	694 (98.7%)	69.5	5,108 (99.1%)
MESA (Ours)	**80.0**	601 (85.5%)	69.6	4,391 (85.2%)

Table 1: Training speed and accuracy of SGD, SAM, SAM's variants, SAF, and MESA on the ImageNet datasets with ResNet-50, ResNet-101, ResNet-152 and ViT-S/32.



(a) Training loss vs Epochs of SAF. (b) The SAM's sharpness measure vs epochs

Figure 3 : (a) SAF and MESA do not affect the convergence of training.
(b) SAF and MESA decrease the sharpness measure of SAM.



Vanilla (SGD) SAM SAF (Ours) MESA (Ours)

Figure 4: Cross-entropy loss landscapes with respect to the Gaussian perturbation (0.07 of weights' norm).

Takeaways: (a) SAF and MESA preserve both SGD's training speed and SAM's performance.
(b) SAF and MESA do not affect the convergence and decrease the sharpness measure of SAM.
(c) SAF and MESA can find flatter minima as SAM does.

Reference:

[1] Sepp Hochreiter and J¨urgen Schmidhuber. Simplifying neural nets by discovering flat minima. In Advances in neural information processing systems, pp. 529–536, 1995.
[2] Laurent Dinh, Razvan Pascanu, Samy Bengio, and Yoshua Bengio. Sharp minima can generalize for deep nets. In International Conference on Machine Learning, pp. 1019–1028. PMLR, 2017.
[3] Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In International Conference on Learning Representations, 2020.
[4] Juntang Zhuang, Boqing Gong, Liangzhe Yuan, Yin Cui, Hartwig Adam, Nicha C Dvornek, James s Duncan, Ting Liu, et al. Surrogate gap minimization improves sharpness-aware training. In International Conference on Learning Representations, 2021.