Generalization and Memorization in Sparse Neural Network

Ziyu Ye Chaoqi Wang Zixin Ding Yuxin Chen

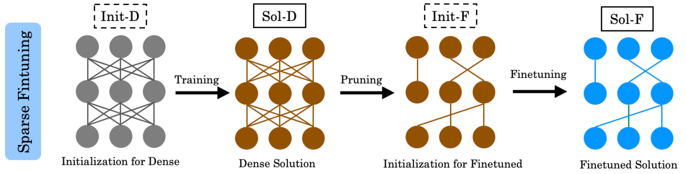
University of Chicago

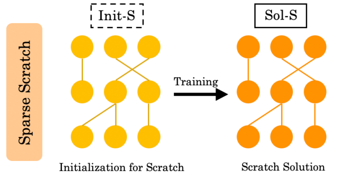
Scan for the latest version.

tinyurl.com/snn-mem

1 *Motivation*: Future is Sparse Training from Scratch

[Backgrounds] True Efficiency by Sparse Training from Scratch

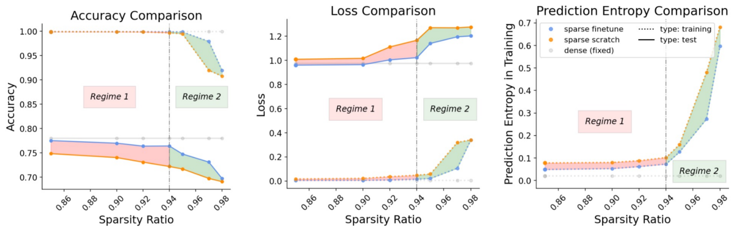




While being efficient, it is challenging to train a sparse network from scratch…

[Challenges] Performance Gap: A Tale of Two Regimes

Performance gap (i.e., *generalization discrepancy*) exists between sparse scratch and sparse finetuning.



[Research Questions] Closing the Gap Requires Understanding

- What is the root cause for the *performance gap* (between sparse scratch and sparse finetuning)?
- How may we close the *performance gap*?

[Preliminaries] Hessian, Jacobian, Fisher Info and Memorization

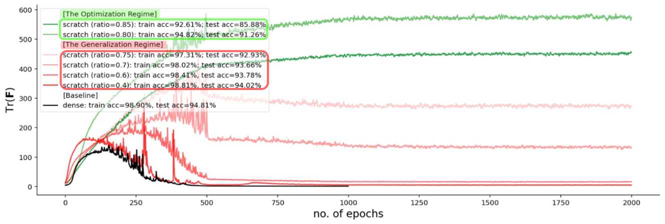
Hessian and the Loss Curvature $\mathbf{H}(\boldsymbol{\theta}) = \nabla^2_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta})$.

Jacobian and Network Sensitivity $\mathbf{J}(\mathbf{x}) = \nabla_{\mathbf{x}^T} f(\mathbf{x}; \boldsymbol{\theta})$.

Fisher Information $\mathbf{F}(\boldsymbol{\theta}) = \mathbb{E}_{\mathbf{x}\sim\mathcal{D}, \hat{y}\sim p_{\boldsymbol{\theta}}(y|\mathbf{x})} \left[\nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\hat{y} \mid \mathbf{x}) \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\hat{y} \mid \mathbf{x})^T \right]$.

Long-Tail Hypothesis of Memorization Memorization of data labels is necessary to achieve good generalization on long-tailed data distribution (Feldman, 2020).

2 *Experiments*: Underlying Mechanisms



	Training Error	Trend of Fisher Information	Reasons for the Generalization Discrepancy
Generalization Regime	near optimal	increases first, then decreases, yet ends up with a higher level compared to sparse finetuning's	sharper minima, higher sensitivity
Optimization regime	far worse than the optimal	increases and hardly decreases	weaker memorization on training data

Table 1: Summary of the underlying mechanisms for sparse training from scratch.

[Generalization Regime] Curse of Information

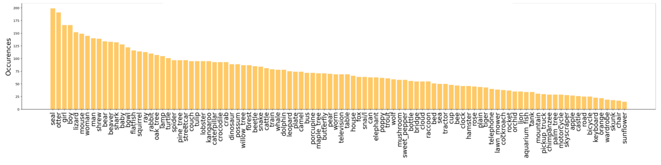
The Curse of Information for Sparse Training:
Sparse scratch requires ***more information*** in learning, hurting its generalization.

	Sparsity Ratio	Tr(F)	Tr(H)	Jacobian Norm
Sol-D	0.00	53.81	66.32	35.60
Sol-F	0.90	543.27	720.49	43.27
	0.94	2573.51	4198.26	52.25
	0.96	4376.32	9618.40	67.92
Sol-S	0.90	2701.68 ↑	10452.93 ↑	63.74 ↑
	0.94	9840.60 ↑	12905.88 ↑	66.36 ↑
	0.96	15818.65 ↑	20715.41 ↑	70.10 ↑

[Optimization Regime] The Memorization Hypothesis

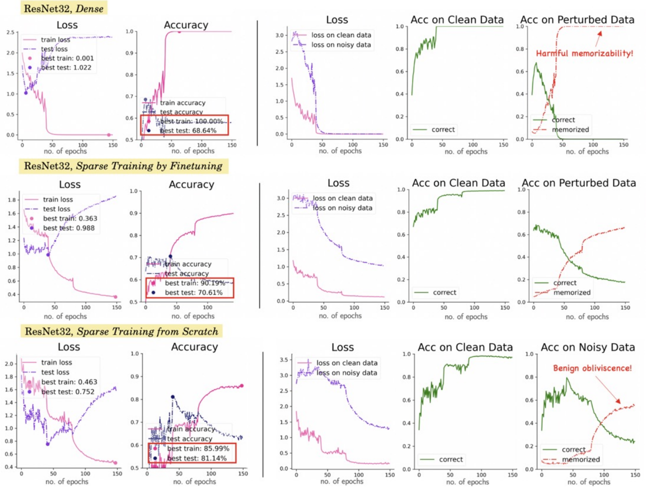
Memorization Hypothesis for Sparse Training:
Sparse scratch are ***weaker at memorization*** than sparse finetuning.

(CIFAR100) It is hard for sparse scratch to distinguish data needs memorization, e.g., that has *hard/low-level/class-specialized features.*



[Train Set] Class Distribution for Data that Sol-S and Sol-F Has Disagreement

[Optimization Regime] Robustness by Memorization



Sparse training from scratch is more robust to label noise. This experiment is conducted on CIFAR-10 with ResNet32. The training set contains 30% perturbed data whose labels are uniformly randomly flipped. The sparsity ratio is 0.95 for sparse training. The learning rate is 0.02 and decay by 0.1 at the 40th and 80th epoch. The left two columns show the performance on the training set (noisy) and test set (clean), and the right two columns show the performance on the clean and noisy data in the training set; the notation correct means predicted label equals to true label, while memorized means predicted label equals to noisy label.

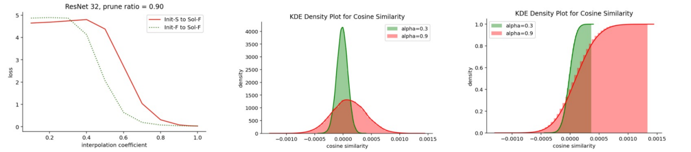
3 *Insights*: Closing the Gap

Regularized **Sparse Training from Scratch**

A naïve approach is to adaptively apply *Fisher information regularization.*

Data-Efficient **Sparse Training from Scratch**

Scheduling data (e.g., iteratively constructing training data subsets) may be effective.



4 Takeaways

- Sparse scratch are weaker at memorization and require more information in learning.
- Next steps: closing the gap for sparse scratch by regularization or data scheduling.