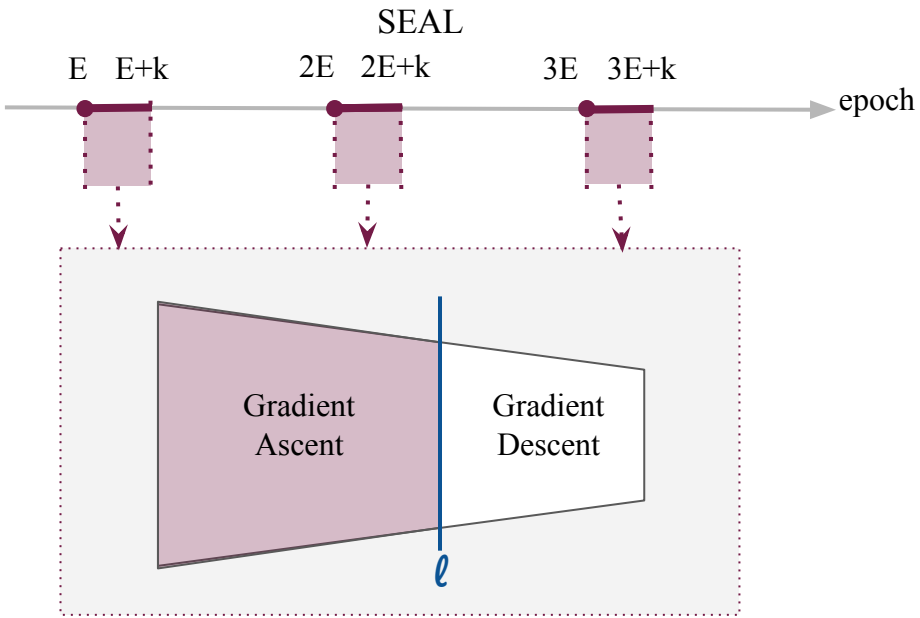
Simulated Annealing in Early Layers Leads to Better Generalization

Amir Sarfi, Zahra Karimpour, Muawiz Chaudhary, Nasir Khalid,
Mirco Ravanelli, Sudhir Mudur, Eugene Belilovsky

Concordia University
Mila – Quebec AI Institute

Our method (SEAL)

- SEAL: Simulated Annealing in Early Layers
- Later layers: Always gradient descent
- Early layers: periodically do gradient ascent for a short time.
  - E.g., every 160 epochs, 40 epochs of ascent and 120 epochs of descent.



SEAL transfer learning

- In-distribution: our method avoids overfitting, surpassing normal training by a large margin.

Method	Tiny-ImageNet
Normal (160 epochs)	54.37
Normal (1,600 epochs)	49.27
SEAL (1,600 epochs)	**59.22**

SEAL transfer learning

- In-distribution: our method avoids overfitting, surpassing normal training by a large margin.
- Our method also outperforms normal training in the transfer learning setting.

Method	Tiny-ImageNet	Flower
Normal (160 epochs)	54.37	34.31
Normal (1,600 epochs)	49.27	26.96
SEAL (1,600 epochs)	**59.22**	**45.68**

Introduction

Iterative Training - Problem Statement

- Imagine we are stuck with the same data for a long period of time.

Challenge:

- The model fits the data well after X epochs. How to train it for 10X epochs?
  - Normal training may lead to overfitting

ResNet-50 on Tiny-ImageNet:

Epoch	Normal
160	54.37
480	51.16
1,600	49.27

Iterative Training - Problem Statement

- Imagine we are stuck with the same data for a long period of time.

Challenge:

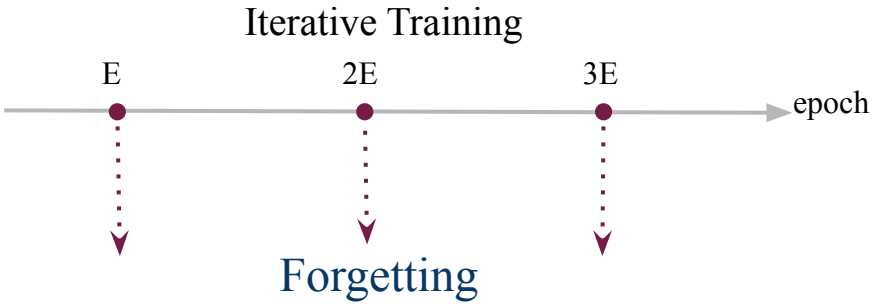
- The model fits the data well after X epochs. How to train it for 10X epochs?

Goal:

- Make best use of the data we already have.

Iterative Training

- Periodically (every E epochs) forget useless information to make room for more training. A period of E epochs is called a “Generation”.
- Forgetting: any operation that lowers the training accuracy.
- **Beginning** of generation: **Forgetting**
  The rest of the generation: **Relearning**



Forget and Relearn: Zhou et al. Fortuitous Forgetting in Connectionist Networks - ICLR 2022

Motivation

Prediction depth (Cont.)

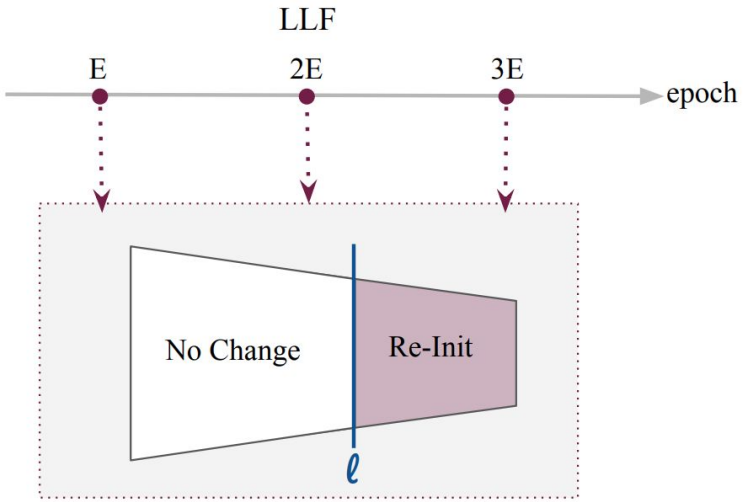
- Prediction depth: For a single sample, the layer after which all layers correctly classify that sample under KNN probe.

- **Better** prediction depth leads to **better**:
  - Uncertainty
  - Confidence
  - Accuracy
  - Speed of learn for that data point

Paper: Baldock et al., Deep learning through the lens of example difficulty - NeurIPS 2021

Later Layer Forgetting - LLF

- **Goal**: improve the prediction depth.
- **Forgetting:**
  - Randomly re-initialize the later layers of the network.
  - Called Later Layer Forgetting (LLF) since it constantly forgets later layers.



Paper: Zhou et al. Fortuitous Forgetting in Connectionist Networks - ICLR 2022

Fortuitous Forgetting - LLF (Cont.)

- LLF enhances in-distribution generalization. (Accs: Tiny-ImageNet)
- But, how does LLF work in Transfer Learning?

ResNet-50

Generation	Normal	LLF
Gen=1	54.37	-
Gen=3	51.16	56.12
Gen=10	49.27	56.92

LLF is bad at Transfer Learning

ResNet-50; Pretrained on Tiny-ImageNet

Method	Tiny-ImageNet	Flower	CUB	Aircraft	MIT	Stanford Dogs
Normal	54.37	34.31	6.49	6.24	25.67	8.99
Normal (long)	49.27	26.96	8.07	6.30	24.85	11.53
LLF	56.92	22.84	5.33	4.65	23.8	8.69

Motivation

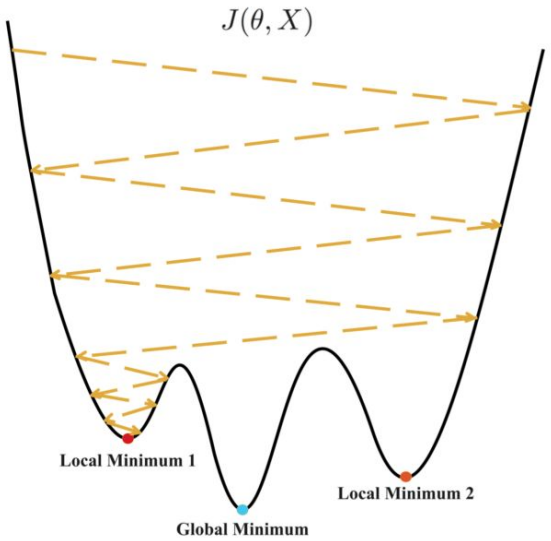
- Goal: We want to have **good prediction depth** + **better transfer learning**.

- Solution: Simulated Annealing in Early Layers!

Methodology & Results

Simulated Annealing in Optimization

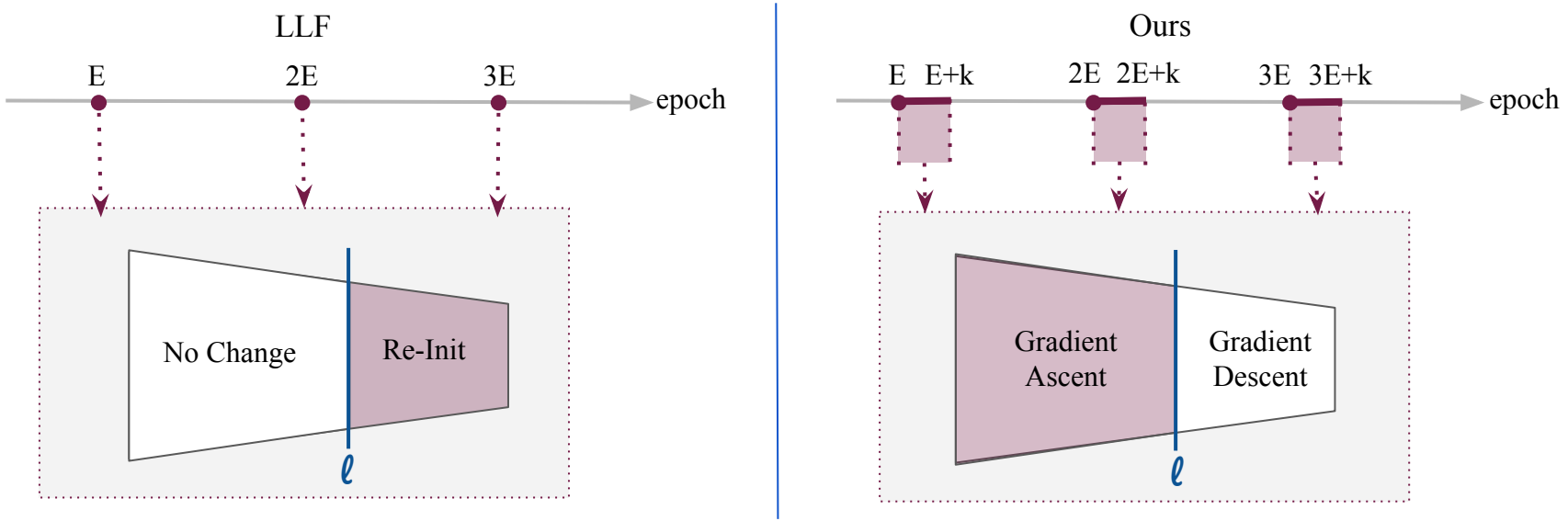
- Randomly perform gradient-ascent with some probability P.
- P is high in the beginning and diminishes at the end.



Paper: Cai, SA-GD: Improved Gradient Descent Learning Strategy with Simulated Annealing

Our method (SEAL)

- Forgetting:
  - For k epochs, perform gradient ascent on early layers of the network



SEAL has much better prediction depth

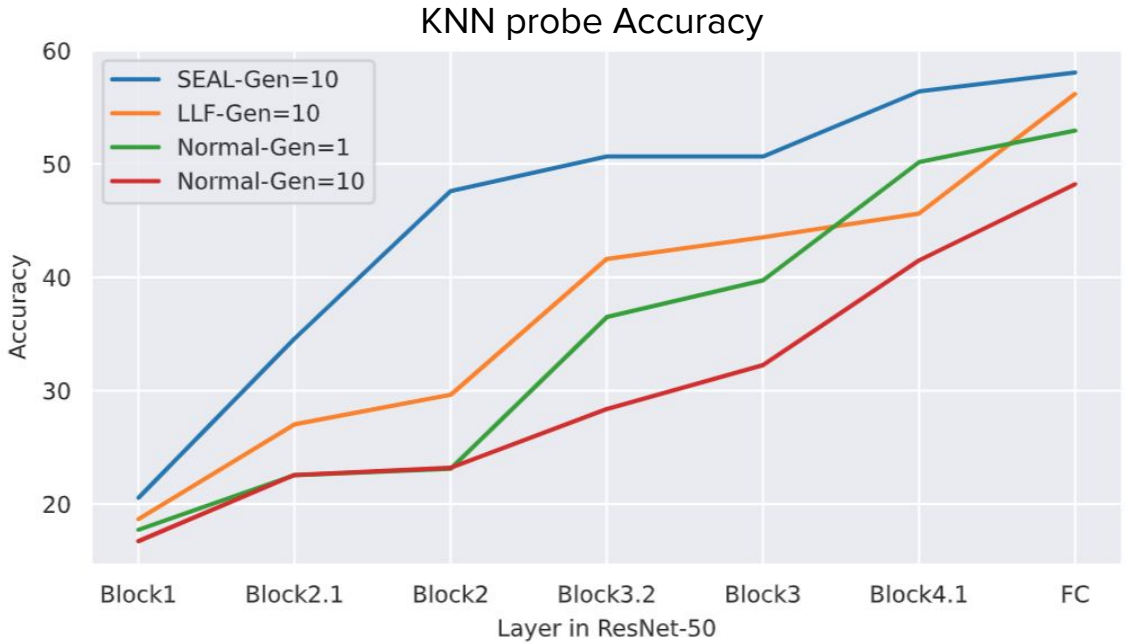


Figure 2. Comparison of layer-wise prediction depth. SEAL gives comparably much stronger predictions early on in the network. Note that block.X.Y denotes the output activations of intermediate layer Y in residual block X. This indicates that our method encourages learning the difficult examples using conceptually simpler and more general features of the early layers. This leads to better overall performance as we progress deeper into the network.

SEAL in-distribution

Generation	Normal	LLF	Ours
Gen=1	54.37	-	-
Gen=3	51.16	56.12	**58.25**
Gen=10	49.27	56.92	**59.22**

Table 2. Comparison of our method with normal training and LLF on Tiny-ImageNet. Please note that the behavior of the first generation for all methods is the same. We significantly outperform standard long training and LLF.

SEAL transfer learning

Method	Tiny-ImageNet	Flower	CUB	Aircraft	MIT	Stanford Dogs
Normal	54.37	34.31	6.49	6.24	25.67	8.99
Normal (long)	49.27	26.96	8.07	6.30	24.85	11.53
LLF	56.92	22.84	5.33	4.65	23.8	8.69
SEAL (Ours)	**59.22**	**45.68**	**8.49**	**9.81**	**35.37**	**12.61**

Table 1. Transferring tiny-imagenet learned features to other datasets using linear probe. Normal, and Normal (long) refer to $G = 1$ and $G = 10$ generations of training, respectively. LLF and SEAL were trained for $G = 10$ generations. Transfer accuracy of LLF after $1,600$ epochs is significantly lower than normal training with both 160 and $1,600$ epochs; our method after $1,600$ epochs surpasses normal training by a large margin. This demonstrates that our method learns much more generalizable features compared to Normal training and LLF.

SEAL: Stronger few-shot transfer learning

- Model: ResNet-50 pretrained on Tiny-ImageNet dataset
- For the target datasets, we only have 20 samples per class (20-shot).

Base Model	EuroSAT	CropDisease
Normal	81.52 ± 1.38	87.43 ± 1.67
Normal (long)	80.03 ± 1.53	89.63 ± 1.53
LLF	46.55 ± 2.42	39.80 ± 2.30
SEAL (ours)	$\mathbf{87.70 \pm 0.52}$	$\mathbf{95.67 \pm 0.28}$

Thank you!