An Empirical Study of Robust Deep Models

Haoye Cai, **Kaidi** Cao, **Bingbin** Liu

Supervised by **Baharan** Mirzasoleiman

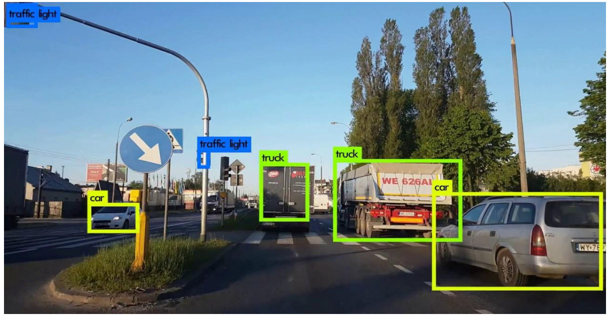
Outline

- Background & Motivation
- Our method: Set Cover & Facility location.
- Application on faster/robust learning
- Experiments on CIFAR10, FashionMNIST.
- Conclusion

Introduction

- "Success" of Deep Learning
  - Capability to fit complex functions by learning from data
  - Greatly improve accuracy for many tasks

Image Classification



Object Detection

Pose Estimation

Introduction

- **Faster** Deep Learning
  - Training deep learning model takes long (GPU) hours
  - Acceleration normally done by large-scale training with distributed systems

- **Robust** Deep Learning
  - Memorization in over-parameterized neural networks can severely hurt generalization in the presence of mislabeled examples
  - Mislabeled examples are to hard avoid in extremely large datasets

Our Goal

We try to develop a training strategy for deep neural nets that:

1. **Faster** training with selected subsets of data
2. **More robust** training by filtering out noisy/harmful data points

We'll address both using our novel data selection & weighting scheme.

Related Work

- **Exploiting the Structure: Stochastic Gradient Methods Using Raw Clusters** [Allen-Zhu et al. 2016]
    - Faster training with the help from the clustering structure of the data
    - Cluster the data points based on their gradient similarity
- **Learning to Reweight Examples for Robust Deep Learning** [Ren et al. 2018]
    - Meta-learning to weight training samples based on gradient directions
    - Need a clean unbiased validation set
- **An Empirical Study of Example Forgetting During Deep NN Learning** [Toneva et al. 2019]
    - Based on forgetting dynamics, a significant fraction of training samples can be omitted w/o hurting performance

Our Method

- **Clustering**
  - Divide the whole dataset into clusters according to a distance metric
  - Data points within a cluster should be similar
  - Two methods: **Set Cover** & **Facility Location**
- **Selection and weighting**
  - Sample one point from each cluster
  - Weight each point according to information within clusters
  - Train (for one epoch) on those selected points

[Recap] Set Cover

Definition: select a sequence of points ordered by number of neighbors.

- Neighbors = points within a ball with radius r.
  - Weighted calculation based on density of the ball.
- Greedy algorithm as an approximation.
- Feature or gradient similarity from pretrained models.

	# clusters	Acc@1	Acc@5
Feature	43105 (38964)	92.26 (91.55)	99.81
Gradient	**43334** (**39715**)	**92.48** (**92.15**)	99.80

Facility Location (FL)

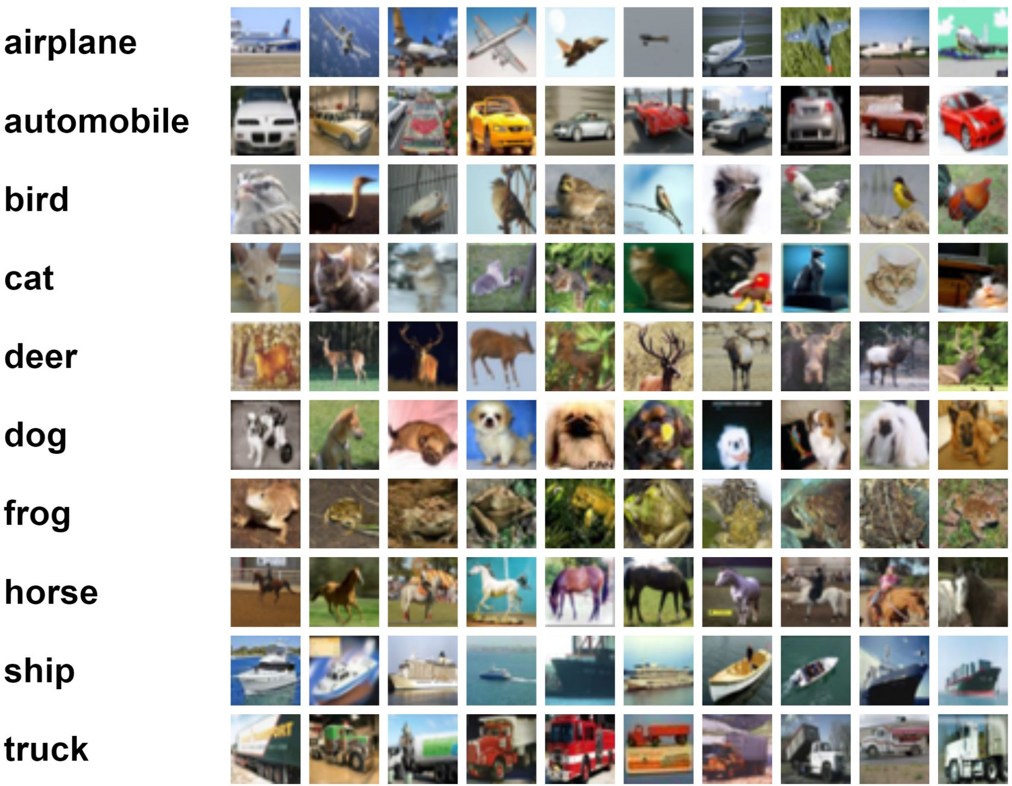
Definition: given a set *D*, select a subset of data points as facilities such that the **total distance to the facilities is minimized**.

- Facilities ~ representatives: best approximate of the total gradient.
- Why: compared to set cover: better globally + no tuning r + faster.
- How: greedy using gradient similarity (L2 distances).
    - Per class clustering: greedy + maintain ordering
    - **Online**: update every each epoch.

Experiments

Dataset: **CIFAR-10**

- 32x32 colour images
- 10 classes
- 50k for training
- 10k for testing



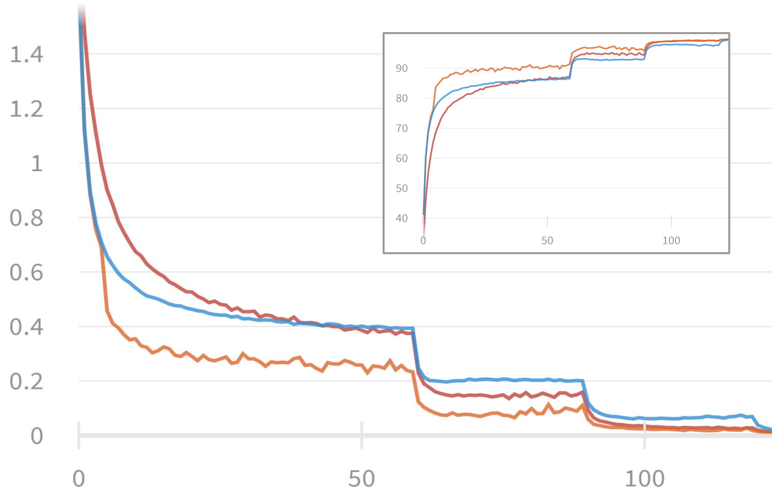
FL - Experiments

Facility ratio	20%	40%	60%	80%	100%
Data covered	40%	60%	80%	92%	100%
Acc@1	80.79	85.17	89.16	91.76	93.08

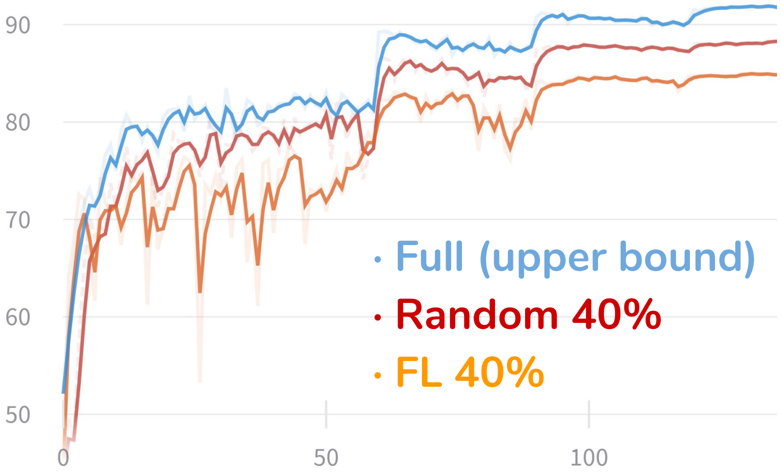
- **Batch size**: 32: differ by 1.4%: 32 > 16 > 64 > 128
- **LR scheduling**: decrease by x0.3 at epoch 60, 90, 120, 160.
  - Others: decrease by x0.1 at epoch 120, 160 (-0.8%); constant small LR (-2%).
- **Optimizer**: SGD worked better than Adam: -1%.
- **Shuffle** the within-class orders: -0.8%; optimizes more slowly.

FL - Analysis

Better training behavior, less helpful on the test set (may be overfitting).



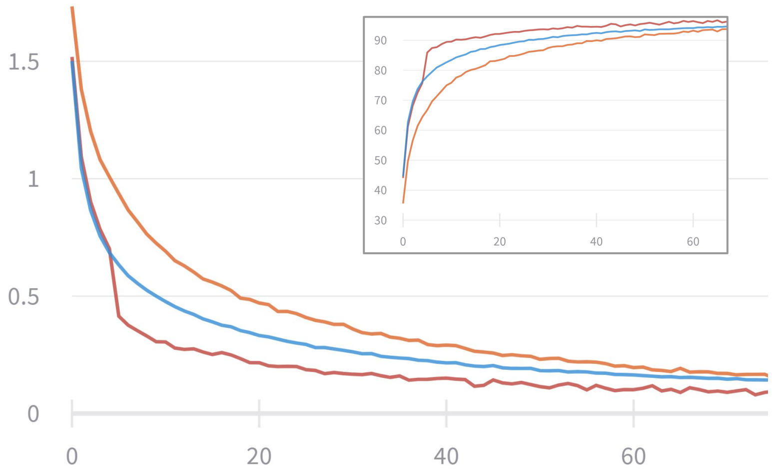
Training



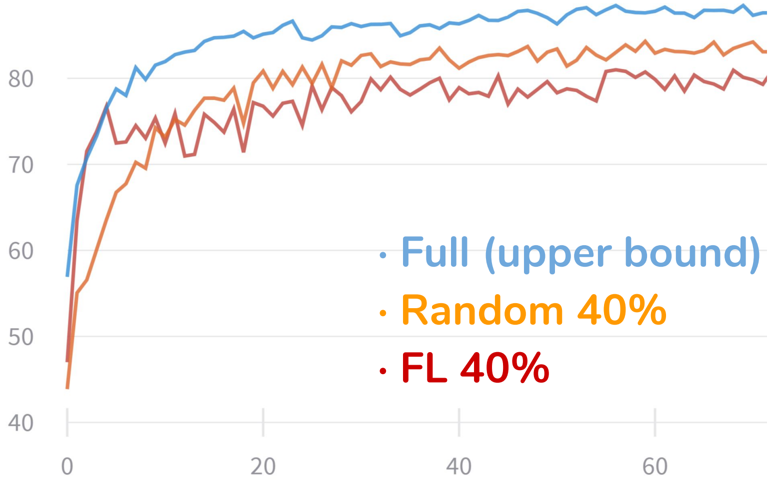
Testing

FL - Analysis

Better training behavior, less helpful on the test set (may be overfitting).



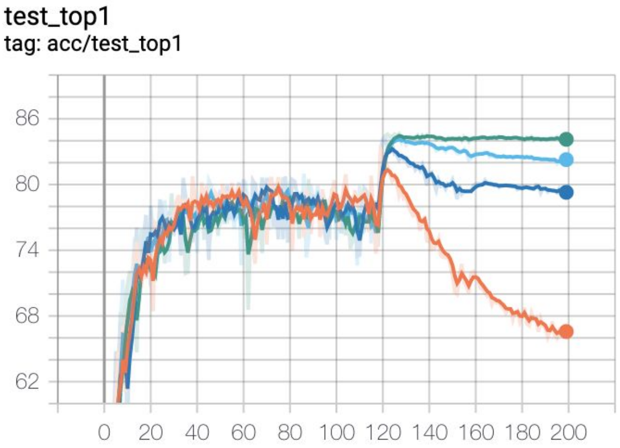
Training

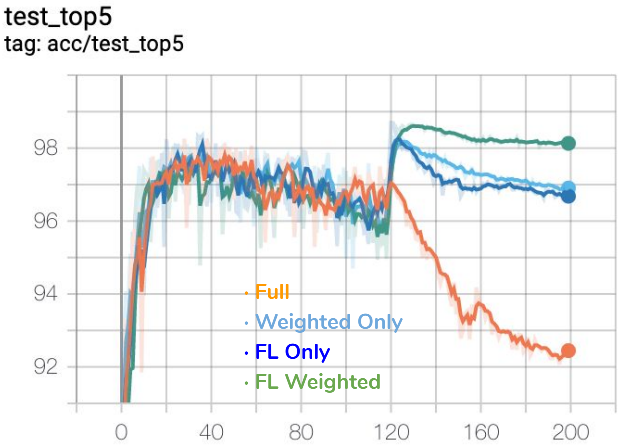


Testing

FL for Robust Training

Comparison: full vs FL only vs weighted only vs FL weighted



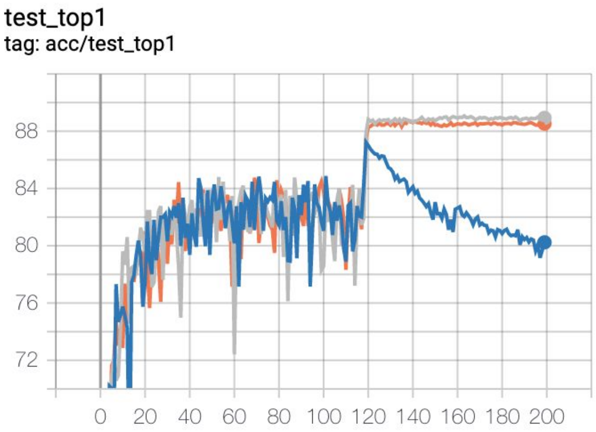


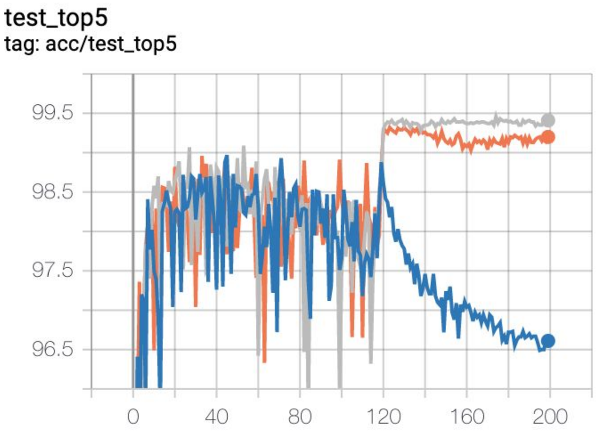
Method	Best Acc
FL weighted	85.16
Weighted only	84.18
FL only	83.64
full	83.20

Noise Ratio = 0.4

FL for Robust Training

Comparison: full vs FL only vs weighted only vs FL weighted



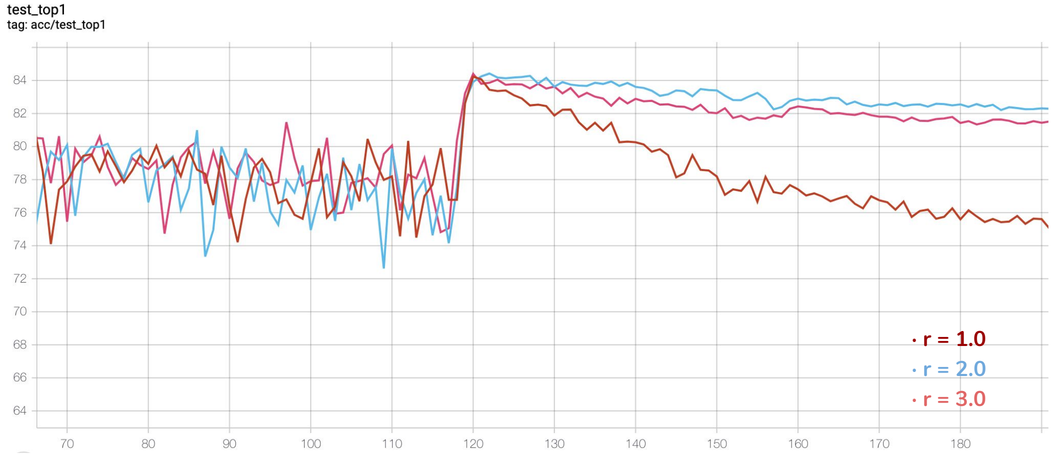


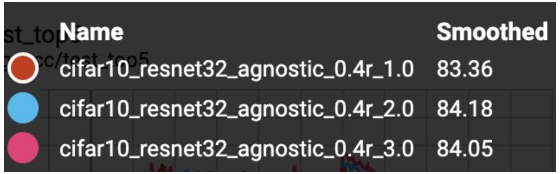
Method	Best Acc
FL weighted	88.73
Weighted only	88.39
full	87.22

Noise Ratio = 0.2

FL for Robust Training

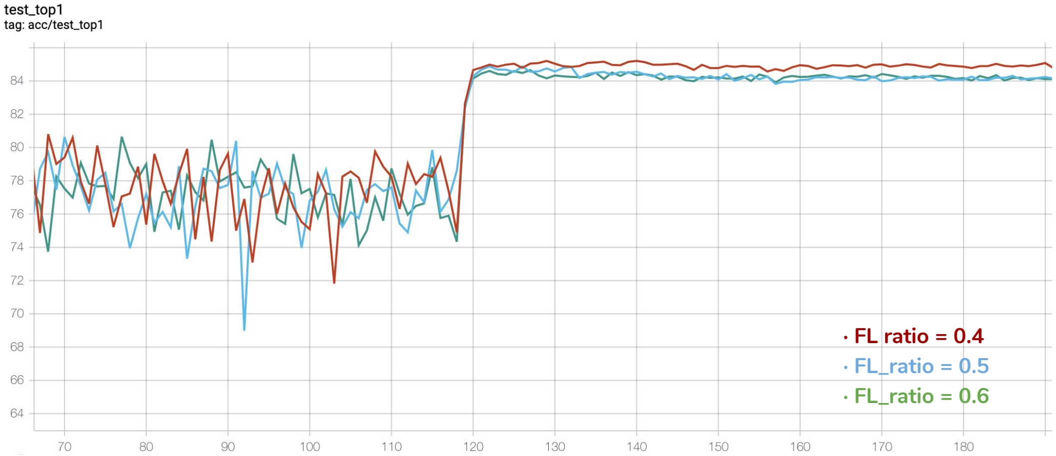
Ablation study on ball radius

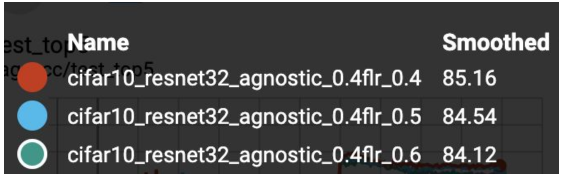




FL for Robust Training

Ablation study on FL ratio

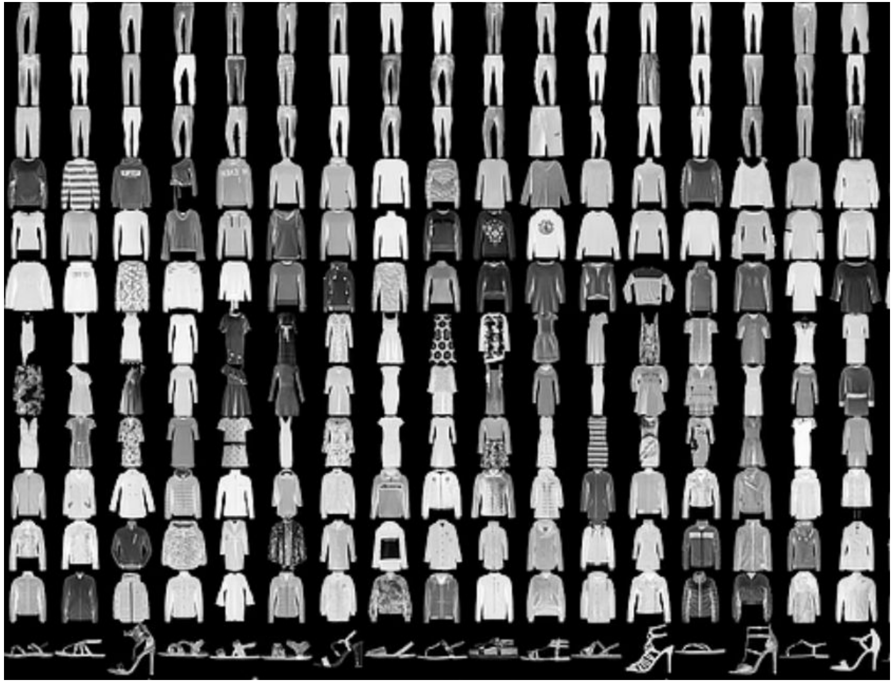




3. Experiments on FashionMNIST

Dataset:

- 28x28 grayscale images
- 10 classes
- 60k for training
- 10k for testing



Results - Faster

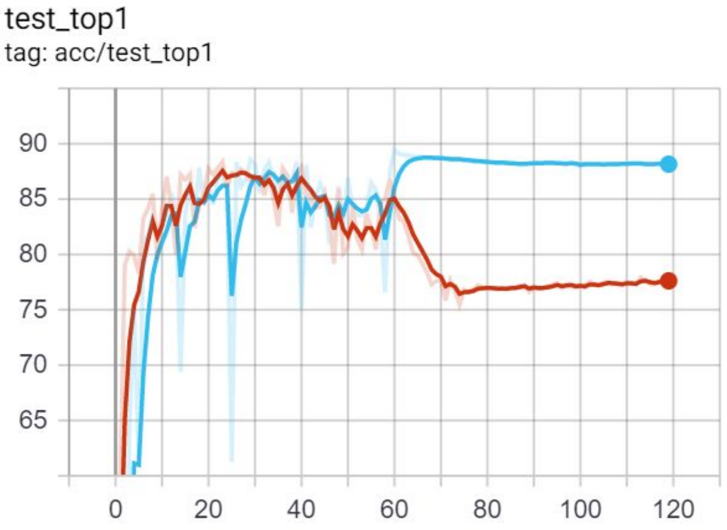
Facility ratio	0.2	0.4	0.6	0.8	1
Data covered	35%	50%	75%	90%	100%
Acc@1	90.68	90.71	91.17	91.45	93.12

- Training much faster (fewer backward passes)
- Use only part of training data
- Small performance compromise

Results - Robust

Comparison: Ours (FL weighted) vs training on the full noisy dataset

➢ Noise ratio: 0.2

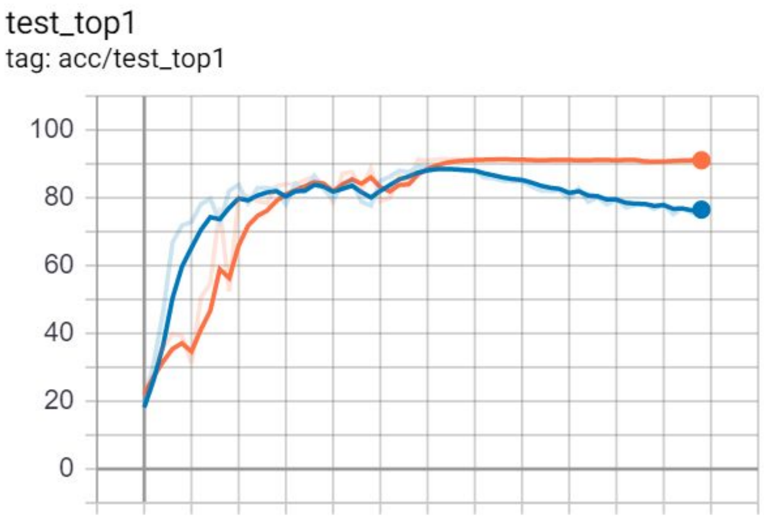


Method	Best Acc
FL weighted	89.47
full	88.35

Results - Robust

Comparison: Ours (FL weighted) vs training on the full noisy dataset

➢ Noise ratio: 0.4



Method	Best Acc
FL weighted	91.42
full	89.26

Conclusion

Project overview: a method for faster & more robust learning:

- Better training behavior at the early stage.
- More robust to noises.

Future directions:

- Closer look at faster training
    - Curriculum learning: gradually increase the training set size.

Thank you!