Paper Sharing Conference of CVPR'2019

Apr 2, 2019 @ Tsinghua

Snapshot Distillation: Teacher-Student Optimization in One Generation

Authors: **Chenglin Yang**, Lingxi Xie, Siyuan Qiao, Alan L. Yuille

Speaker: **Lingxi Xie**

Highlights

- What are we doing?
  - Generic network optimization
- Who can benefit from this work?
  - Any work related to network optimization
- Is this approach complicated?
  - No, it can be implemented with a few lines of code

Outline

- Introduction: Teacher-Student Optimization
- Why Teacher-Student Optimization Works?
- How to Accelerate Teacher-Student Optimization?
- Conclusion and Future Work

Outline

- Introduction: Teacher-Student Optimization
- Why Teacher-Student Optimization Works?
- How to Accelerate Teacher-Student Optimization?
- Conclusion and Future Work

Deep Learning and Network Optimization

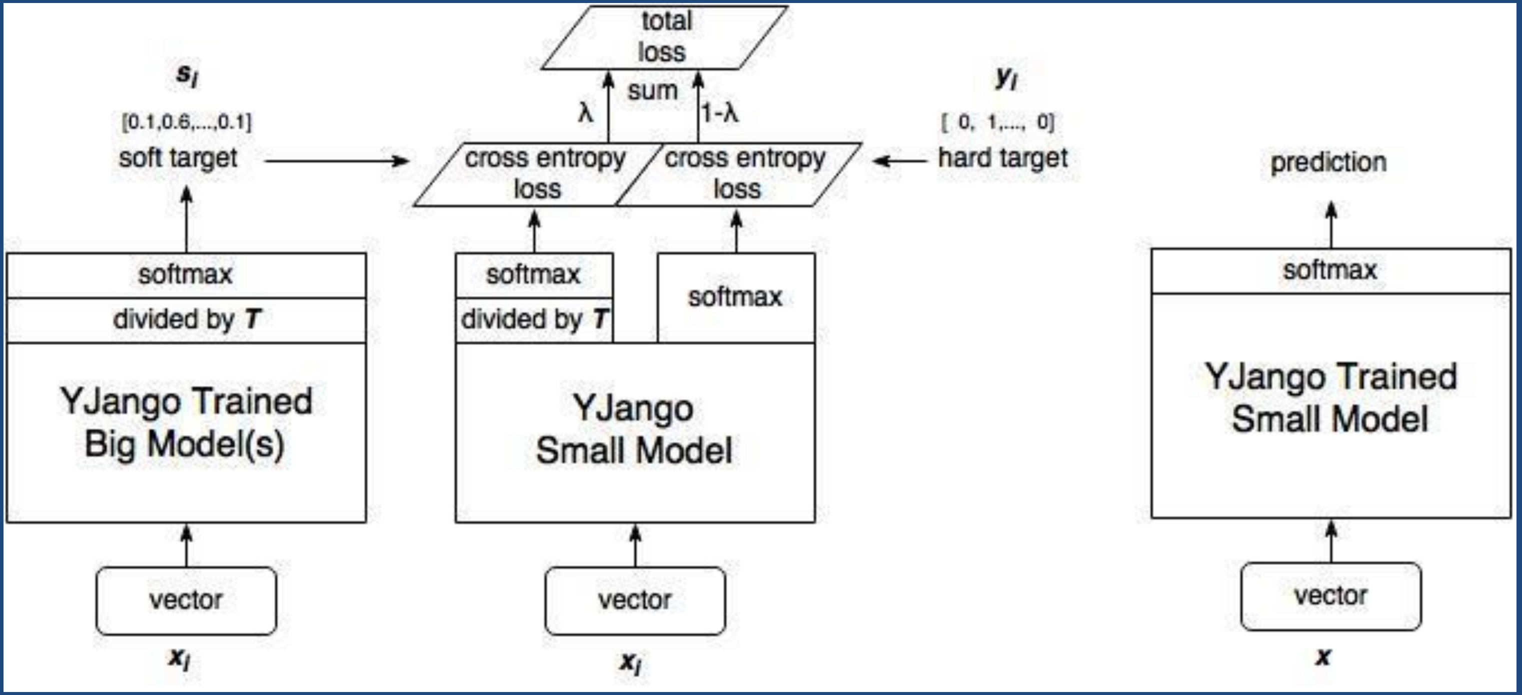
- Two key components of a deep neural network
  - Backbone: AlexNet, VGGNet, GoogLeNet, ResNet, DenseNet, SENet, *etc.*
  - Loss function: cross-entropy, *etc.*
- Mini-batch-based optimization in the context of deep learning

$$\mathcal{L}(\mathcal{B};\theta)=\frac{1}{|\mathcal{B}|}\sum_{(\mathbf{x}_n,\mathbf{y}_n)\in\mathcal{B}}\{-\mathbf{y}_n^{\mathrm{T}}\ln \boldsymbol{F}(\mathbf{x}_n;\theta)\}$$

$\mathbf{x}_n$: input image;	$\mathbf{y}_n$: output label;	$\mathcal{B}$: mini-batch;
$\boldsymbol{F}(\mathbf{x}_n;\theta)$: model;	θ: parameters;	$\mathcal{L}(\mathcal{B};\theta)$: loss function

Teacher-Student Optimization

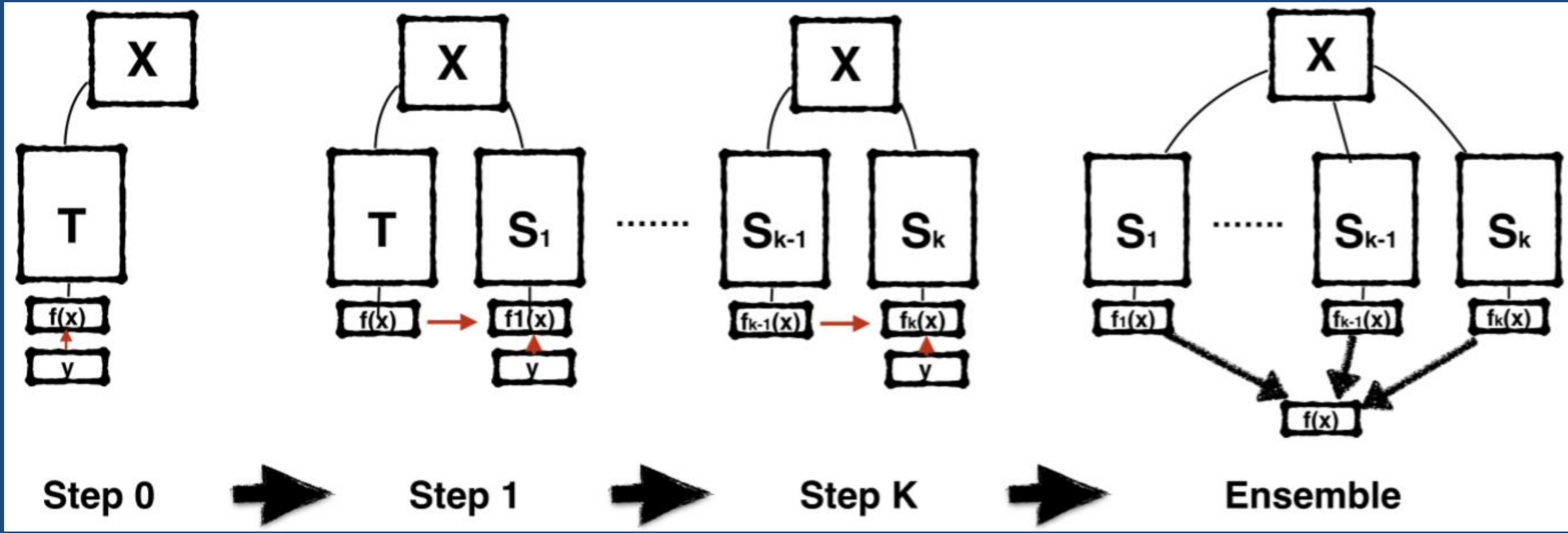
- In the early age, it was used to compress a neural network [KD]
  - Core idea: using a deeper (teacher) model to assist training a shallower (student) model



[KD] **G. Hinton** *et al.*, Distilling the Knowledge in a Neural Network, *NIPS workshop*, 2014.

Teacher-Student Optimization (cont.)

- Since 2018, researchers started using this idea for training better models [BAN]
  - Teacher and student models are equally complex, but the student gets better trained



[BAN] **T. Furlanello** *et al.*, Born Again Neural Networks, *ICML*, 2018.

Outline

- Introduction: Teacher-Student Optimization
- Why Teacher-Student Optimization Works?
- How to Accelerate Teacher-Student Optimization?
- Conclusion and Future Work

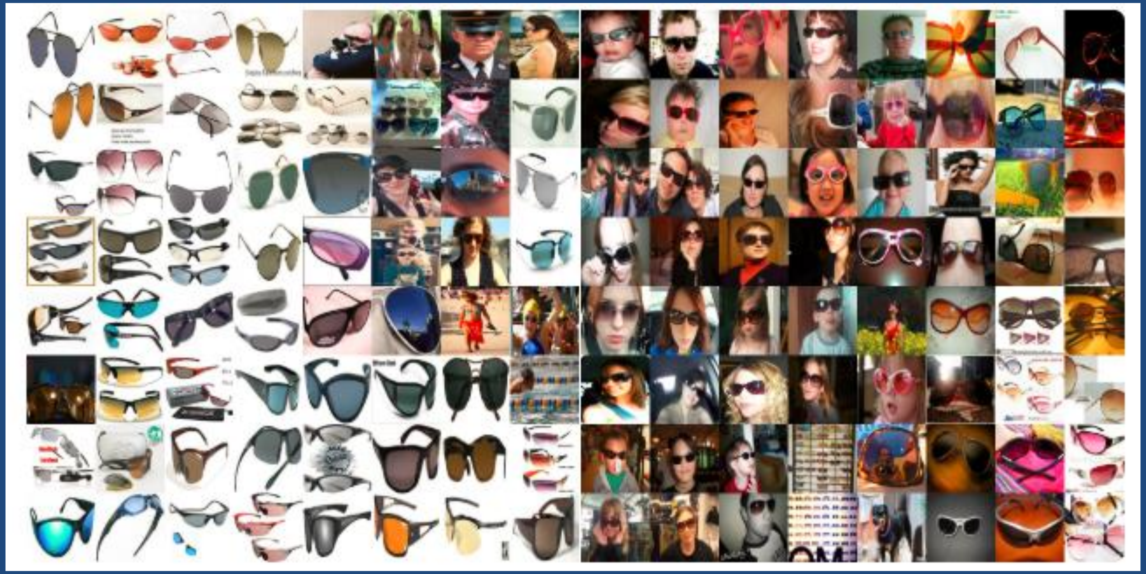
Why Teacher-Student Optimization Works?

- A plausible explanation on why teacher-student optimization works
  - "Work": improving the performance of a model under the same complexity
  - Both KD and BAN explained it as a kind of "dark knowledge", which "carries information on the similarity between output categories"
- Our work [TT] discusses this problem in details, and develops a new way of enhancing such "dark knowledge"

[KD] **G. Hinton** *et al.*, Distilling the Knowledge in a Neural Network, *NIPS workshop*, 2014.
[BAN] **T. Furlanello** *et al.*, Born Again Neural Networks, *ICML*, 2018.
[TT] **C. Yang** *et al.*, Training Deep Neural Networks in Generations: A More Tolerant Teacher Educates Better Students, *AAAI*, 2019.

Part of Dark Knowledge is Secondary Information

- Secondary information
  - When an image (or any sample in general) is classified into a class, a small fraction of scores or probability is assigned to other classes – they deliver useful knowledge



All these images are labeled "sunglasses"

"croquet ball"

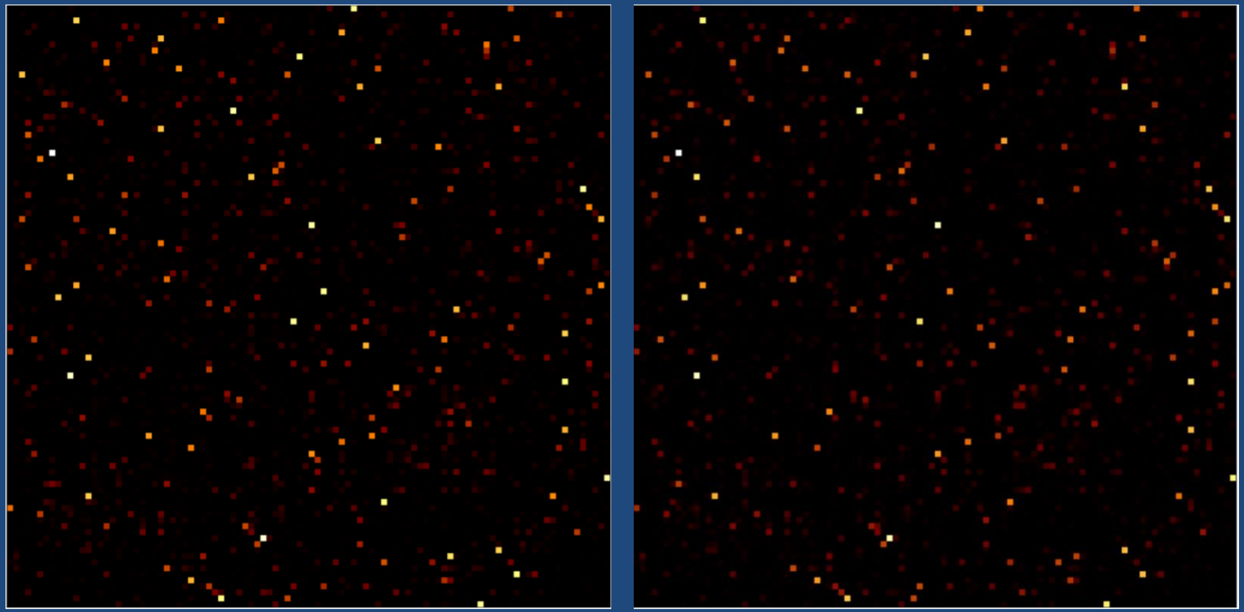
[TT] **C. Yang** *et al.*, Training Deep Neural Networks in Generations: A More Tolerant Teacher Educates Better Students, *AAAI*, 2019.

Part of Dark Knowledge is Secondary Information

- What is the best teacher?
  - A strict (one-hot) teacher? No!
  - A more tolerant teacher works better

	Top-1	Top-2	Top-3	Top-4	Train	Test
Gen #0	99.28	0.57	0.09	0.03	99.74	71.55
Gen #1	98.68	1.00	0.18	0.06	99.63	71.41
Gen #2	98.42	1.13	0.23	0.09	99.60	72.30
Gen #3	98.33	1.19	0.24	0.09	99.62	72.26
Gen #4	98.28	1.24	0.25	0.09	99.59	72.52

Confidence distribution (%) on top-4 classes, obtained in a born-again process (with 4 more generations). The dataset is CIFAR100, and the network is ResNet-110.



Confusion matrices of the first teacher model and the first student model (row: the ground-truth class, column: the 2nd highest class, yellow indicates large value).

[TT] **C. Yang** *et al.*, Training Deep Neural Networks in Generations: A More Tolerant Teacher Educates Better Students, *AAAI*, 2019.

How a "Tolerant Teacher" Works

- We deliberately reduce the confidence of the first teacher model
  - Original teacher optimization

$$\mathcal{L}(\mathcal{B};\theta)=\frac{1}{|\mathcal{B}|}\sum_{(\mathbf{x}_n,\mathbf{y}_n)\in\mathcal{B}}\left\{-\mathbf{y}_n^{\mathrm{T}}\ln\boldsymbol{F}(\mathbf{x}_n;\theta)\right\}$$

  - Modified teacher optimization

$$\mathcal{L}(\mathcal{B};\theta)=\frac{1}{|\mathcal{B}|}\sum_{(\mathbf{x}_n,\mathbf{y}_n)\in\mathcal{B}}\left\{-\eta\cdot\mathbf{y}_n^{\mathrm{T}}\ln\boldsymbol{F}(\mathbf{x}_n;\theta)+(1-\eta)\cdot\left[f_{a_1}-\frac{1}{K-1}\sum_{k=2}^{K}f_{a_k}\right]\right\}$$

η: mixing parameter; f_{a_k}: the k-th largest score

[TT] **C. Yang** *et al.*, Training Deep Neural Networks in Generations: A More Tolerant Teacher Educates Better Students, *AAAI*, 2019.

Results on CIFAR100

- 300 epochs per generation, 5 + 1 generations

	Gen #0	Gen #1	Gen #2	Gen #3	Gen #4	Gen #5
Baseline (100 layers)	22.20 (22.89)	–	–	–	–	–
$\mathfrak{D}(0.6, 0.6)$	23.96 (25.00)	21.29 (21.34)	**20.51** (21.59)	20.83 (20.99)	21.01 (21.53)	21.27 (21.61)
+Ensemble	–	20.20	18.38	17.79	17.37	**17.25**
$\mathfrak{D}(0.7, 0.6)$	22.98 (23.43)	21.24 (21.50)	21.48 (21.80)	**20.94** (21.47)	21.51 (21.69)	21.87 (22.28)
+Ensemble	–	19.63	18.83	17.70	17.56	**17.23**
Baseline (190 layers)	17.22 (17.62)	–	–	–	–	–
$\mathfrak{D}(0.6, 0.6)$	18.87 (19.40)	17.42 (17.99)	17.26 (18.00)	17.13 (17.52)	17.24 (17.75)	**17.01** (17.22)
+Ensemble	–	16.83	15.94	15.43	**15.18**	15.21
$\mathfrak{D}(0.7, 0.6)$	18.63 (19.12)	17.44 (17.78)	**16.72** (17.21)	16.89 (16.98)	17.39 (17.71)	17.24 (17.41)
+Ensemble	–	16.37	15.20	15.11	14.93	**14.47**

(Zhang et al. 2017b)	19.25	(Huang et al. 2017a)	17.40	(Han, Kim, and Kim 2017)	17.01
(Zhang et al. 2017a)	16.80	(Gastaldi 2017)	15.85	(Furlanello et al. 2018)	14.90

[TT] C. Yang *et al.*, Training Deep Neural Networks in Generations: A More Tolerant Teacher Educates Better Students, *AAAI*, 2019.

Results on ImageNet

- ResNet-18
- 90 epochs per generation
- 5 + 1 generations

	Gen #0		Gen #1		Gen #2		Gen #3		Gen #4		Gen #5	
Baseline	30.50	11.07	—	—	—	—	—	—	—	—	—	—
$\mathfrak{D}(0.6, 0.6)$	32.52	11.23	30.28	10.23	30.12	10.15	29.92	10.25	29.77	10.19	**29.60**	**10.11**
+Ensemble	—	—	30.01	9.98	28.94	9.53	28.51	9.36	28.23	9.28	**28.08**	**9.23**

[TT] C. Yang *et al.*, Training Deep Neural Networks in Generations: A More Tolerant Teacher Educates Better Students, *AAAI*, 2019.

Outline

- Introduction: Teacher-Student Optimization
- Why Teacher-Student Optimization Works?
- How to Accelerate Teacher-Student Optimization?
- Conclusion and Future Work

How to Accelerate Teacher-Student Optimization?

- Conventional teacher-student optimization is too slow!
- How to solve this issue?
  - Finishing the entire teacher-student optimization within one generation!

A One-Generation Flowchart

- Core idea: using a previous snapshot as the teacher

Algorithm 1: Snapshot Distillation

Input : training set $\mathcal{D}$, number of iterations L,
training configurations $\{\gamma_l, \lambda_l^{\mathrm{T}}, \lambda_l^{\mathrm{S}}, c_l\}_{l=1}^{L}$;

Initialize $\boldsymbol{\theta}_0$;
for $l = 1, 2, \ldots, L$ **do**
    Sample a mini-batch $\mathcal{B}_l$ from $\mathcal{D}$;
    Compute loss $\mathcal{L}(\mathcal{B}_l; \boldsymbol{\theta}_{l-1})$ using Eqn (3);
    $\boldsymbol{\theta}_l \leftarrow \boldsymbol{\theta}_{l-1} - \gamma_l \cdot \nabla_{\boldsymbol{\theta}_{l-1}} \mathcal{L}(\mathcal{B}_l; \boldsymbol{\theta}_{l-1})$
end
Return: $\mathbb{M} : \mathbf{y} = \mathbf{f}(\mathbf{x}; \boldsymbol{\theta} = \boldsymbol{\theta}_L)$.

$$\mathcal{L}(\mathcal{B}_l; \boldsymbol{\theta}_{l-1}) = -\frac{1}{|\mathcal{B}_l|} \sum_{(\mathbf{x}_n, \mathbf{y}_n) \in \mathcal{B}_l} \{\lambda_l^{\mathrm{S}} \cdot \mathbf{y}_n^{\top} \ln \mathbf{f}(\mathbf{x}_n; \boldsymbol{\theta}_{l-1}) + \lambda_l^{\mathrm{T}} \cdot \mathrm{KL}[\mathbf{f}(\mathbf{x}_n; \boldsymbol{\theta}_{c_l}) \,\|\, \mathbf{f}(\mathbf{x}_n; \boldsymbol{\theta}_{l-1})]\}.$$

λ_l: Adaptive weights of teacher (T) and student (S)
γ_l: the learning rate
c_l: the iteration number when the snapshot is taken

[SD] **C. Yang** *et al.*, Snapshot Distillation: Teacher-Student Optimization in One Generation, *CVPR*, 2019.

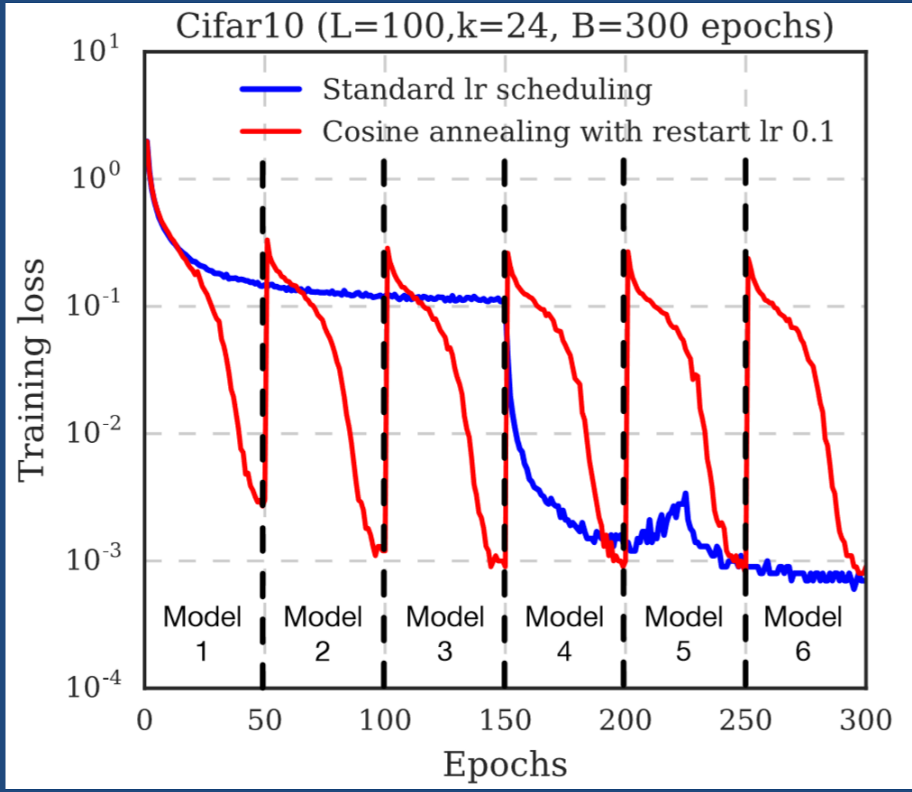
Principles and Solution

- Three principles
  - #1: high quality of the teacher model: a good teacher guarantees reliable supervision
  - #2: teacher and student shall be sufficiently different: otherwise, the impact of the teacher signal becomes rather weak
  - #3: secondary information: this is based on our previous observations
- Solution: cyclic, cosine-annealing learning-rates with asymmetric distillation
  - Each local minimum corresponds to a teacher (#1)
  - A jump in the learning rate guarantees sufficient difference (#2)
  - At each teacher signal, divide the *logits* by a temperature of T (#3)

[SD] **C. Yang** *et al.*, Snapshot Distillation: Teacher-Student Optimization in One Generation, *CVPR*, 2019.

Connections to Snapshot Ensemble (SE)

- Snapshot ensemble: training a single model with a few cycles of learning rate annealing, and obtaining good performance with the ensemble of multiple snapshots
- If we switch off teacher-student optimization, SD will degenerate to SE
- By adding teacher-student optimization, SD achieves better performance than SE in either single models or model ensemble



[SE] **G. Huang** *et al.*, Snapshot Ensembles: Train 1, Get M for Free, *ICLR*, 2017.
[SD] **C. Yang** *et al.*, Snapshot Distillation: Teacher-Student Optimization in One Generation, *CVPR*, 2019.

Results on CIFAR100

- 300 epochs
- 4 mini-generations

Backbone	Alg.	T	$\mathbb{M}_{\#L_1}$	$\mathbb{M}_{\#L_2}$	$\mathbb{M}_{\#L_3}$	$\mathbb{M}_{\#L_4}$	*best*	*ensemble*
ResNet20	**BL**	N/A	–	–	–	33.57	33.57	–
	SE	N/A	36.17	33.36	32.98	32.66	32.54	30.86
	SD	2	36.17	33.78	32.98	32.31	32.31	32.08
	SD	3	36.17	33.69	32.24	**31.97**	**31.76**	**30.76**
ResNet32	**BL**	N/A	–	–	–	31.61	31.61	–
	SE	N/A	33.78	32.15	31.41	30.74	30.51	28.93
	SD	2	33.78	32.07	31.05	30.67	30.57	29.80
	SD	3	33.78	31.52	30.64	**30.32**	**30.16**	**28.71**
ResNet56	**BL**	N/A	–	–	–	30.23	29.94	–
	SE	N/A	32.85	31.60	30.45	29.68	29.55	27.93
	SD	2	32.85	30.47	29.72	**29.29**	**29.22**	28.11
	SD	3	32.85	30.82	29.55	29.37	29.28	**27.74**
ResNet110	**BL**	N/A	–	–	–	28.77	28.53	–
	SE	N/A	31.89	29.81	29.07	28.27	28.09	26.45
	SD	2	31.89	29.84	28.71	**27.71**	**27.52**	27.19
	SD	3	31.89	29.22	28.37	27.87	27.75	**26.19**
DenseNet100	**BL**	N/A	–	–	–	22.49	22.00	–
	SE	N/A	24.31	22.76	22.16	22.18	22.00	**19.63**
	SD	2	24.31	23.10	22.06	21.78	21.59	20.27
	SD	3	24.31	23.19	21.60	**21.17**	**21.17**	19.71
DenseNet190	**BL**	N/A	–	–	–	16.82	16.69	–
	SE	N/A	18.98	18.12	16.95	**16.84**	16.70	**15.70**
	SD	2	18.98	17.48	16.32	18.02	**16.06**	15.72
	SD	3	18.98	17.67	16.95	18.65	16.33	15.92

[SD] **C. Yang** *et al.*, Snapshot Distillation: Teacher-Student Optimization in One Generation, *CVPR*, 2019.

Results on ImageNet and Beyond

- ImageNet
  - 90 epochs
  - 2 mini-generations
- PASCALVOC07 (det)
  - Faster R-CNN
- PASCALVOC12 (seg)
  - DeepLab-v3

Backbone	Alg.	$\mathbb{M}_{\#L_1}$		$\mathbb{M}_{\#L_2}$	
		Top-1	Top-5	Top-1	Top-5
ResNet101	**BL**	–	–	21.62	5.80
ResNet101	**SE**	22.94	6.51	22.14	6.07
ResNet101	**SD**	22.94	6.51	**21.25**	**5.55**
ResNet152	**BL**	–	–	21.17	5.66
ResNet152	**SE**	22.56	6.44	21.84	5.84
ResNet152	**SD**	22.56	6.44	**20.93**	**5.55**

Backbone	mAP @ 2007	mIOU @ 2012
ResNet152-**BL**	73.49	77.53
ResNet152-**SD**	**74.93**	**77.97**

[SD] C. Yang *et al.*, Snapshot Distillation: Teacher-Student Optimization in One Generation, *CVPR*, 2019.

Outline

- Introduction: Teacher-Student Optimization
- Why Teacher-Student Optimization Works?
- How to Accelerate Teacher-Student Optimization?
- Conclusion and Future Work

Conclusions and Future Work

- Teacher-student optimization is useful
  - Model compression and acceleration
  - Model optimization and regularization
- Teacher-student optimization needs to be better explained
  - Why it works: this is what we have done, but needs more exploration
- Teacher-student optimization can be associated with other methods
  - Acceleration: this is what we have done, but needs more exploration
  - Applications 1: incremental learning – we have a submission to ICCV 2019
  - Applications 2: neural architecture search – we have an ongoing work

Thanks

- Questions, please?