GENERATIVE TEACHING NETWORKS: ACCELERATING NEURAL ARCHITECTURE SEARCH BY LEARNING TO GENERATE SYNTHETIC TRAINING DATA

报告人：陈宇航

时间：2020年8月6日

Outline

- Motivation
- Method
- Search space
- Experiments

motivation

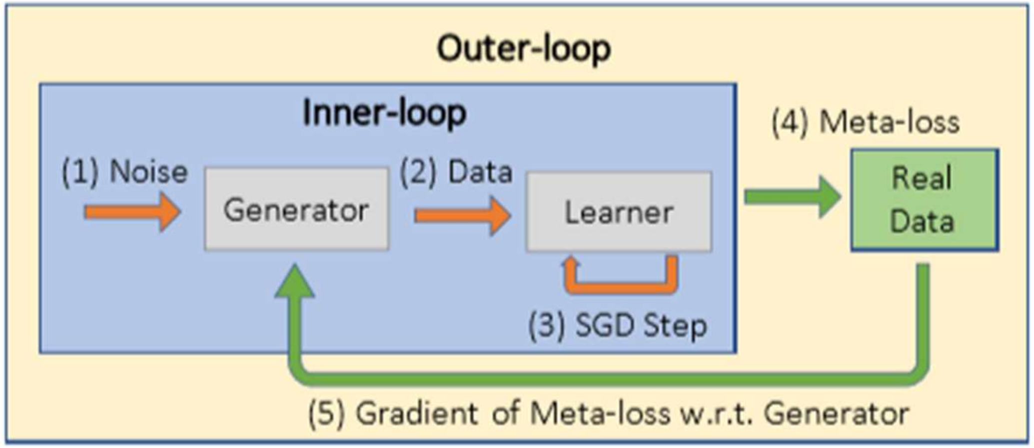
This paper investigates the question of whether we can train a data-generating network that can produce synthetic data that effectively and efficiently teaches a target task to a learner

Accelerate the evaluation of candidate architectures for neural architecture search (NAS),

Outline

- Motivation
- Method
- Search space
- Experiments

Method



(a) Overview of Generative Teaching Networks

1 generate data from noise

2 The learner is trained to perform well on the generated data

3 The trained learner is then evaluated on the real training data in the outer-loop to compute the outer-loop meta-loss

4 The gradient of the generator parameters are computed

Method

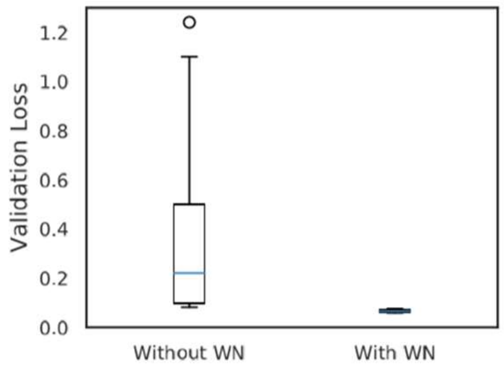
Weight Normalization

$$W = g * \frac{V}{\|V\|}$$

update scalar g and vector V
reason:

Batch normalization normalizes the forward propagation of activations in a long sequence of parameterized operations

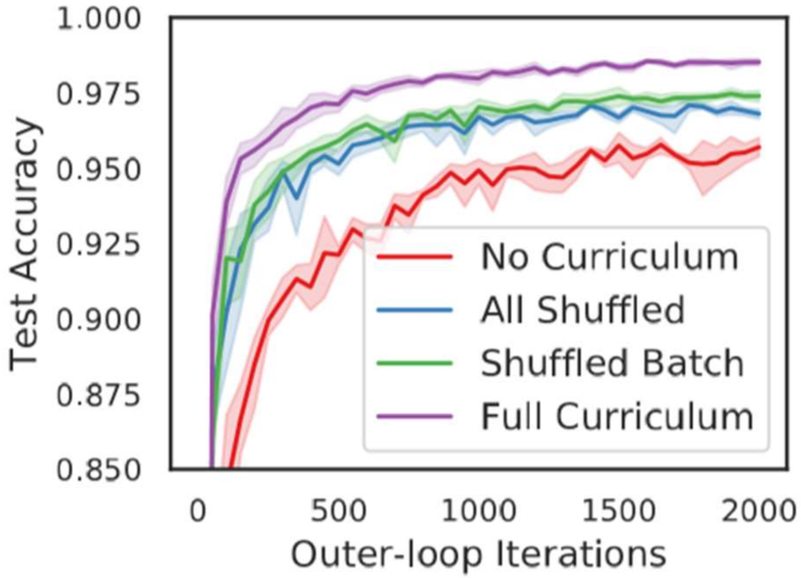
In meta-gradient training both the activations and weights result from along sequence of parameterized operations and thus both should be normalized.



(b) GTN stability with WN

Method

Curriculum



1) sampling the noise vector z for each sample independently from a Gaussian distribution(No Curriculum)

2) learns a fixed set of 4,096 input samples that are presented in a random order(GTN - All Shuffled)

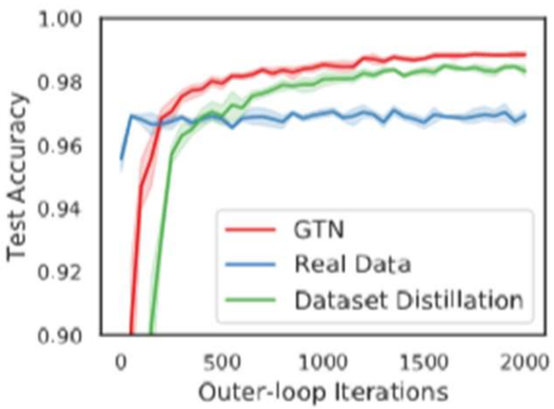
3) learns 32 batches of 128 samples each, the order in which the batches are presented is randomized (GTN-Shuffled Batch)

4) learns a deterministic sequence of 32 batches of 128 samples (GTN-Full Curriculum)

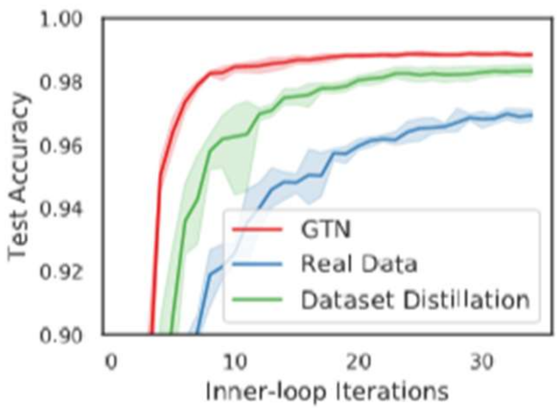
Outline

- Motivation
- Method
- Experiments

Experiment



(a) Meta-training curves



(b) Training curves

(c) GTN-generated samples

1) Real data:Training learners with random mini-batches of real data, as is ubiquitous in SGD
2) Dataset Distillation: Training learners with synthetic data, where training examples are directly encoded as tensors optimized by the meta-objective
3) GTN - Our method where the training data presented to the learner is generated by a neural network

Experiment

(a) CIFAR10 inner-loop training (b) CIFAR10 GTN samples (c) CIFAR10 correlation

optimize our GTN ahead of time using proxy learners in p6000GPU(8h)

1) 128 steps with GTN vs 100epoch with real data is 0.3606
2) the top 50% of architectures is 0.5582
3) 3 epochs with real data is 0.5235, 9x cost-reduction per trained model

Experiment

Model	Error(%)	#params	GPU Days
Random Search + GHN (Zhang et al., 2018)	4.3 ± 0.1	5.1M	0.42
Random Search + Weight Sharing (Luo et al., 2018)	3.92	3.9M	0.25
Random Search + Real Data (baseline)	3.88 ± 0.08	12.4M	10
Random Search + GTN (ours)	$\mathbf{3.84} \pm 0.06$	8.2M	0.67
Random Search + Real Data + Cutout (baseline)	3.02 ± 0.03	12.4M	10
Random Search + GTN + Cutout (ours)	$\mathbf{2.92} \pm 0.06$	8.2M	0.67
Random Search + Real Data + Cutout (F=128) (baseline)	2.51 ± 0.13	151.7M	10
Random Search + GTN + Cutout (F=128) (ours)	$\mathbf{2.42} \pm 0.03$	97.9M	0.67

Experiment

Model	Error(%)	#params	Random	GPU Days
NASNet-A (Zoph & Le, 2017)	3.41	3.3M	✗	2000
AmoebaNet-B + Cutout (Real et al., 2019)	2.13	34.9M	✗	3150
DARTS + Cutout (Liu et al., 2018b)	2.83	4.6M	✗	4
NAONet + Cutout (Luo et al., 2018)	2.48	10.6M	✗	200
NAONet-WS (Luo et al., 2018)	3.53	2.5M	✗	0.3
NAONet-WS + Cutout (Luo et al., 2018)	2.93	2.5M	✗	0.3
ENAS (Pham et al., 2018)	3.54	4.6M	✗	0.45
ENAS + Cutout (Pham et al., 2018)	2.89	4.6M	✗	0.45
GHN Top-Best + Cutout (Zhang et al., 2018)	2.84 ± 0.07	5.7M	✗	0.84
GHN Top (Zhang et al., 2018)	4.3 ± 0.1	5.1M	✓	0.42
Random-WS (Luo et al., 2018)	3.92	3.9M	✓	0.25
Random Search + Real Data (baseline)	3.88 ± 0.08	12.4M	✓	10
RS + Real Data + Cutout (baseline)	3.02 ± 0.03	12.4M	✓	10
RS + Real Data + Cutout (F=128) (baseline)	2.51 ± 0.13	151.7M	✓	10
Random Search + GTN (ours)	$\mathbf{3.84} \pm 0.06$	8.2M	✓	0.67
Random Search + GTN + Cutout (ours)	$\mathbf{2.92} \pm 0.06$	8.2M	✓	0.67
RS + GTN + Cutout (F=128) (ours)	$\mathbf{2.42} \pm 0.03$	97.9M	✓	0.67

The End!