Adversarial Network Compression

Radek Bartyzal

Let's talk ML in Prague

18. 4. 2018

Prior work

Ensembles

Diverse models with similar validation performances can be often be combined to achieve predictive power superior to each of the constituent models. [3]

Born again trees

Learn a single tree that is able to recover the performance of a multiple-tree predictor. [4]

Knowledge distillation = model compression

Transfer knowledge acquired by a learned teacher model to a new simpler student model. [5]

Knowledge distillation

Teacher

- high-capacity model
- good performance

Student

- more compact model
- not as good performance as the teacher but better than if it was trained without it

By transferring knowledge, one hopes to benefit from the student's compactness while suffering only minimal degradation in performance.

Knowledge distillation

Teacher produces soft targets = probabilities of incorrect classes = the key to generalization outside of the training dataset.

Training student = minimize weighted average of:

- cross entropy with the soft targets
- cross entropy with the hard targets = labels

Knowledge distillation results

System	Test Frame Accuracy	WER
Baseline	58.9%	10.9%
10xEnsemble	61.1%	10.7%
Distilled Single model	60.8%	10.7%

Figure: DNN acoustic models used in Automatic Speech Recognition. Distilled model trained by an ensemble of models performs better than the baseline. [5]

Generative Adversarial Networks

- Generator G tries to create images that look real = approximate train data distribution
- Discriminator D tries to distinguish real images from G's images

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))].$$

Figure: Training procedure of GAN. [2]

Adversarial Network Compression

- teacher is trained on hard labels and then frozen
- student trained only on soft targets = no labels
- teacher and student have the same architecture
- architectures are CNN ResNets
- using GAN discriminator to align the teacher-student distributions
- last layer features (before logits) are given to D = richer signal than logits
- L2 loss of logits from student VS teacher forces student to mimic teacher

Logits

Unscaled log-probability values = before the softmax activation function.

Adversarial Network Compression

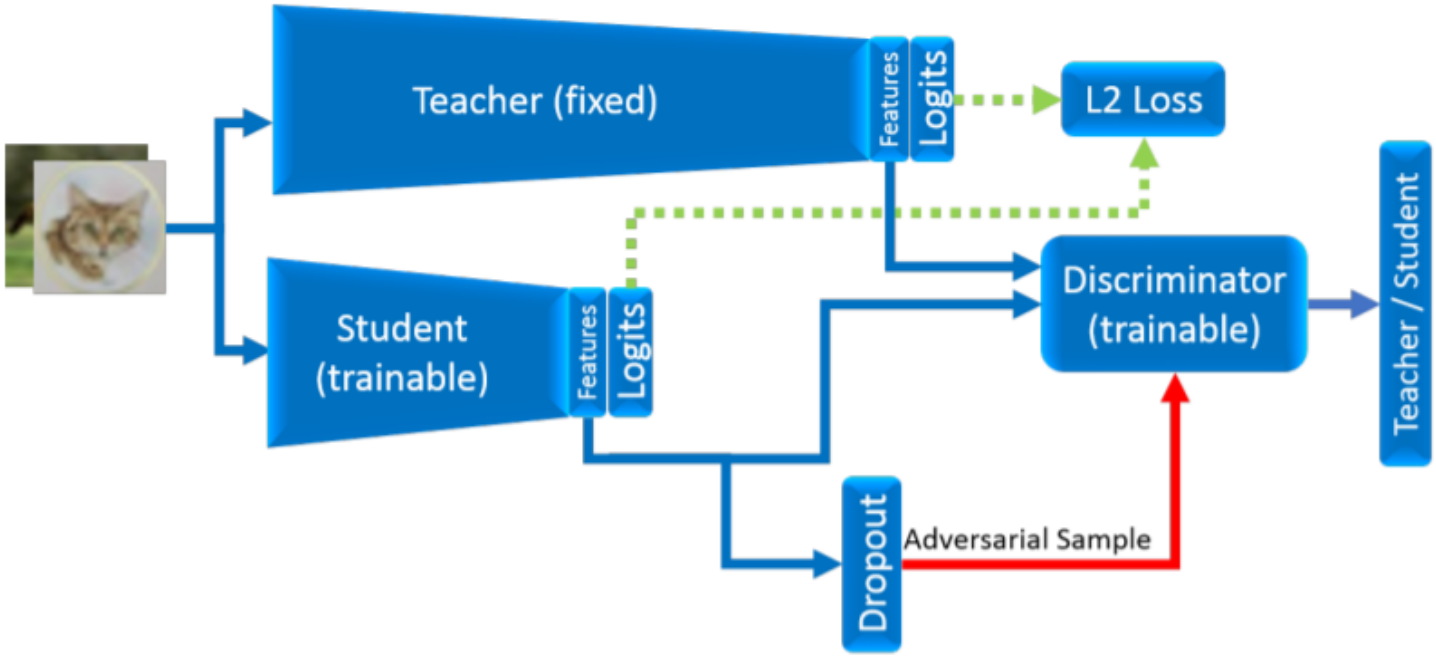


Fig. 1. Adversarial Network Compression: Our method consists of the *teacher*, *student* and discriminator D networks. The teacher is trained in advance and used for supervision during adversarial learning, while the *student* and D are both trainable. D takes as input the features from the *teacher* and *student*, as well as, the adversarial sample (i.e. student labeled as teacher). For the adversarial sample, we empirically found that dropout is beneficial. In addition, there is a L2 loss to force the *student* to mimic the output of the *teacher*.

Training

- $f_t^k(x)$, $f_s^l(x)$ = teacher, student features for input x
- $p_{teacher}(x)$ and $p_{student}(x)$ = feature distributions
- noise input z = dropout in the student

$$\mathcal{L}_{Adv}(f_s, D) = \mathbb{E}_{f_t^k(x) \sim p_{teacher}(x)}[\log(D(f_t^k(x)))] + \mathbb{E}_{f_s^l(x) \sim p_{student}(x), z \sim p_z(z)}[\log(1 - D(f_s^l(x)))].$$

$$\mathcal{L}_{Data}(f_s) = \mathbb{E}_{f_s(x) \sim p_{student}(x)}[\| f_t(x) - f_s(x) \|^2].$$

$$\arg\min_{f_s} \max_{D} \mathcal{L}_{Adv}(f_s, D) + \lambda \mathcal{L}_{Data}(f_s) + \mathcal{L}_{regul}(D)$$

Figure: L_{data} forces student to exactly copy $f_t(x)$ not just the distribution $p_{teacher}$.

Regularization of Discriminator

Goal is to prevent the discriminator from dominating the training, while retaining its capacity:

- Standard L1, L2: No guarantee that the D will become weaker w.r.t student.
- (New) **Adversarial samples for D:** Update D with student samples labeled as teacher's. (discriminator is normally updated only with correct labels)
  = longer game between student and D
  = more gradient updates for student
  + same adversarial samples to update both student and D

Regularization of Discriminator

Dataset	Teacher	Student	W/o Regul.	L1	L2	Ours
CIFAR-10	ResNet-164	ResNet-20	10.07	8.19	8.16	8.08
CIFAR-100	ResNet-164	ResNet-20	34.10	33.36	33.02	32.45
SVHN	ResNet-164	ResNet-20	3.73	3.67	3.68	3.66
F-MNIST	NiN	LeNet-4	9.62	8.91	8.75	8.64

Figure: Adversarial samples for D clearly outperform standard regularization techniques on tested tasks.

Evaluation on CIFAR

Table 3. CIFAR-10 Evaluation We evaluate the components of our approach. ResNet-164 Parameters: **2.6M**, FLOPs: **97.49B**. ResNet-20 Parameters: **0.27M**, FLOPs: **10.52B**. Our *student*, ResNet-20, has around 10x less parameters.

Model	Error[%]
Supervised *teacher* ResNet-164	6.57
Supervised *student* ResNet-20	8.58
Our *student* (D with logits)	8.31
+ dropout in *student*	8.10
Our *student* (D with features)	8.10
+ dropout in *student*	8.08

Table 4. CIFAR-100 Evaluation The component evaluation is presented. We use the same *teacher* and *student* models as in CIFAR-10. ResNet-164 Parameters: **2.6M**, FLOPs: **97.49B**. ResNet-20 Parameters: **0.27M**, FLOPs: **10.52B**.

Model	Error[%]
Supervised *teacher* ResNet-164	27.76
Supervised *student* ResNet-20	33.36
Our *student* (D with logits)	33.96
+ dropout in *student*	33.41
Our *student* (D with features)	33.40
+ dropout in *student*	32.45

Evaluation on CIFAR

CIFAR-10	Error[%]	Param.	**CIFAR-100**	Error[%]	Param.
L2 - Ba *et al.* [10]	9.07	0.27M	Yim *et al.* [55]	36.67	-
Hinton *et al.* [12]	8.88	0.27M	FitNets [56]	35.04	2.50M
Quantization [20]	8.87	0.27M	Hinton *et al.* [12]	33.34	0.27M
FitNets [56]	8.39	2.50M	L2 - Ba *et al.* [10]	32.79	0.27M
Binary Connect [23]	8.27	15.20M	Our *student*	**32.45**	0.27M
Yim *et al.* [55]	11.30	-			
Our *student*	**8.08**	0.27M			

Figure: Comparison with methods in related works.

Evaluation on ImageNet

Model	Top-1 Error[%]	Top-5 Error[%]	Parameters
Supervised *teacher* (ResNet-152)	27.63	5.90	58.21M
Supervised *student* (ResNet-50)	30.30	10.61	37.49M
XNOR [6] (ResNet-18)	48.80	26.80	13.95M
Binary-Weight [6] (ResNet-18)	39.20	17.00	13.95M
L2 - Ba *et al.* [10] (ResNet-18)	33.28	11.86	13.95M
MobileNets [32]	29.27	10.51	4.20M
L2 - Ba *et al.* [10] (ResNet-50)	27.99	9.46	37.49M
Our *student* (ResNet-18)	32.89	11.72	13.95M
Our *student* (ResNet-50)	**27.86**	**8.76**	37.49M

Figure: ResNet-50 clearly benefits from from knowledge distillation.

Residual Networks (ResNet)

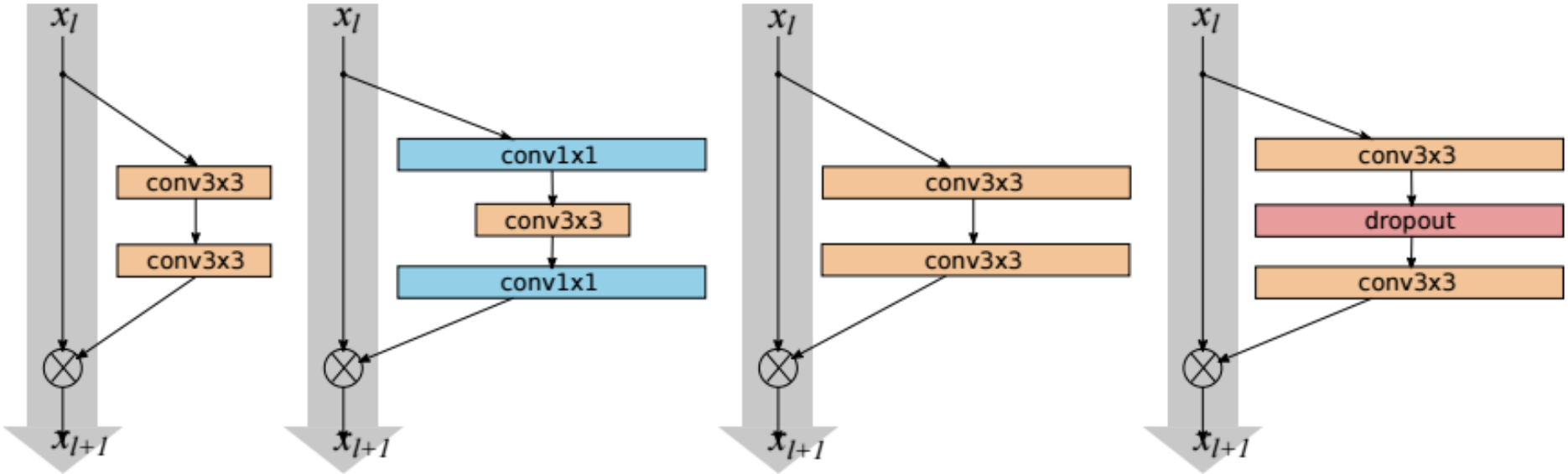
- Add skip-connection that bypasses the non-linear transformations with an identity function.
- Identity function and the output of previous layer are combined by summation, which may impede the information flow in the network.
- "Thin and deep" = small number of filters, 1000+ layers

Diminishing feature reuse

Gradient flowing through skip connections is not forced to go through residual block weights $\Longrightarrow$

- few blocks learn useful representations
- many blocks share very little information with small contribution to the final goal

ResNet: Residual block



(a) basic (b) bottleneck (c) basic-wide (d) wide-dropout

Figure: Various residual blocks used in the paper. Batch normalization and ReLU precede each convolution (omitted for clarity). [**?**]

Sources

1. Belagiannis, Vasileios, Azade Farshad, and Fabio Galasso. "Adversarial Network Compression." arXiv preprint arXiv:1803.10750 (2018). Accessible from: https://arxiv.org/abs/1803.10750

2. Goodfellow, Ian, et al. "Generative adversarial nets." Advances in neural information processing systems. 2014. Accessible from: http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf

3. Hansen, Lars Kai, and Peter Salamon. "Neural network ensembles." IEEE transactions on pattern analysis and machine intelligence 12.10 (1990): 993-1001.

Sources

4. Breiman, Leo, and Nong Shang. "Born again trees." ps (1996). Accessible from: https://www.stat.berkeley.edu/~breiman/BAtrees.pdf

5. Hinton, Geoffrey, Oriol Vinyals, and Jeff Dean. "Distilling the knowledge in a neural network." arXiv preprint arXiv:1503.02531 (2015). Accessible from: https://arxiv.org/pdf/1503.02531.pdf