Knowledge distillation using ensemble of student networks

Swapnil Dewalkar
Department of Computer Science
IIT-Hyderabad
cs17mtech11004@iith.ac.in

Uttaran Sinha
Department of Computer Science
IIT-Hyderabad
cs17mtech11003@iith.ac.in

Tarun Patidar
Department of Computer Science
IIT-Hyderabad
cs17mtech11025@iith.ac.in

Abstract

Traditional neural networks usually focus on accuracy over space and time constraints. But this cripples the deployment of such networks on memory and power constrained devices. Even though some work has been done to train smaller "student" networks, the idea of using an ensemble of such students has not been visited much. In this paper we propose a way to use the idea to train multiple students to attain better compression, parallelism and accuracy over traditional single students.

1 Introduction:

The idea of using an ensemble of small classifiers, such as SVMs or random forests instead of a large model is not ground-breaking. But using an idea of smaller neural networks in unprecedented as deep neural nets often get the job done. But real time classification is still a challenge since it takes quite some time and memory to store and execute an model. Our paper assumes that we have no bound over computation time and space during training, but such constraints appear while testing. We show that using only a handful of small classifiers is enough to gain respectable accuracy when compared to the "teacher" model. We can also exploit parallelism as all students can run independently and each use very few parameters (about 1/100) of the teacher.

2 Dataset:

2.1 MNIST:

We first test on the classic MNIST dataset where the state-of-the-art "teacher" network obtains over 96% accuracy.

2.2 CIFAR-10

Next we test on the CIFAR-10 dataset where there is a larger scope of compression and parallelism as compared to MNIST.

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

3 Methodology:

3.1 Overview:

1. Train Teacher: We trained a state-of-the-art teacher model to "teach" the student networks.
2. Train Single Student: We trained one student, smaller than the teacher and try to mimic the teacher.
3. Train Ensembles: Lastly, we train a bunch of even smaller student networks and combine their knowledge to mimic the teacher and hopefully surpass the performance of the single student network.

3.2 Procedure:

- We are taking the softmax outputs of the pre-trained teacher network and the actual labels of the data.
- We take mini-batches of the input data, feed them to the students and teacher, and try to match the teacher softmax outputs on the students.
- We take a modified softmax output of the teacher (by using temperature methodology) and the actual labels of the data. We do a weighted sum of the two loss functions and train our students.
- Finally, while testing, we take the average prediction of all the students and display the prediction as overall output.

4 Experiment Setup:

4.1 Architecture of MNIST:

- Teacher
  - CONV1 32
  - CONV2 64
  - FC 1024 units
  - Max pooling 2*2
- Student
  - CONV1 8
  - CONV2 8
  - FC 256 units
  - Max pooling 2*2
  - Accuracy: 70-80
- Ensemble
  - Variation:
    - CONV1 [2,4,6,8,16]
    - Filter [2,3,4,5,6]
    - FC [32,64,128,256]
    - Max pooling 2*2
  - 100 Small student
  - Selected top 16 skipping 256 and 128 in FC layer and 2, 16 in Convolution layers for evaluation.

4.2 Architecture of CIFAR-10:

- Teacher
  - CONV1 and CONV2 32
  - Pooling and Dropout with 0.2
  - CONV3 and CONV4 64

  - Pooling and Dropout with 0.6
  - CONV6 and CONV6 128
  - Pooling and Dropout with 0.2
  - FC 2048 units
- Student
  - CONV1 and CONV2 8
  - Pooling and Dropout with 0.2
  - CONV3 and CONV4 16
  - Pooling and Dropout with 0.3
  - CONV6 and CONV6 32
  - Pooling and Dropout with 0.4
  - FC 512 units
- Ensemble
  - Variation:
    - CONV [4,8,16,32]
    - Filter [3,5]
    - FC [32,64,128]
  - 18 Small student

5 Evaluation :

5.1 Teacher:

For teacher we have cross-entropy as the loss function and it is evaluated based on the accuracy in the test set.

5.2 Student:

For students, we have two loss functions while training.
1) The difference between softmax layer values (with temperature) using cross-entropy loss
2) The actual difference between the prediction and the true label (one-hot vector) via cross-entropy.

$$Loss = \lambda * H(y_{True}, P_{student}) + (1 - \lambda) * H(P_{Teacher}^{Temperature}, P_{Student}^{Temperature})$$

where, H is Cross-Entropy Loss

Student is evaluated based on the accuracy in the test set

5.3 Ensemble:

We combine the students' outputs and count the most common prediction and evaluated based on the accuracy in the test set.

6 Result:

6.1 MNIST:

Note: Used Gradient Descent

Model	Accuracy	Parameters	Steps
MNIST-Teacher	96.87 %	3,213,664	25,000
MNIST-Student	95.2 %	100,752	20,000
MNIST-Smaller Student	70-80 % each	-	10,000
MNIST-Ensemble (16 best small students)	87.64 %	190,880	10,000
MNIST-Ensemble (24 best small students)	89.76%	443,120	10,000

6.2 CIFAR-10:

Note: Used Adam Optimizer

Model	Accuracy	Parameters	Steps
CIFAR-10-Teacher	80 %	4,194,304	20,000
CIFAR-10-Student	60	149,040	6000
CIFAR-10-Smaller Student	50-60% each	-	3000
CIFAR-10-Ensemble (24 student)	62 %	1,329,360	3000

7 Observation:

- We have noticed that training more students help with accuracy.
- Accuracy increases when the value of Temperature is increased.
- Number of epochs improve accuracy upto a certain limit after which it saturates.
- Ensemble of students dramatically improve the accuracy of the entire model when compared to single small student predictions.
- Best results for MNIST are seen when Temp = 10 Lambda = 0.75 Kernel Size = 5x5 No of feature maps = 8, FC units = 128

8 Future work:

Our current work can be extended to other datasets like CIFAR-100 and Imagenet. We also want to explore the possibility of training an even larger ensemble of even smaller students.
We can emphasize on using boosting methods instead of max voting.

9 References:

[1] Geoffrey Hinton, Oriol Vinyals, Jeff Dean "Distilling the Knowledge in a Neural Network "
[2] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, Yoshua Bengio "FitNets: Hints for Thin Deep Nets"