DO DEEP CONVOLUTIONAL NETS REALLY NEED TO BE DEEP AND CONVOLUTIONAL?

Wojciech Ratusznik, Taariq Nazar, Kiar Fatah, Andreas Westman

Abstract

In this project, we challenge the hypothesis presented [1] of whether deep convolutional networks have to be deep and convolutional in order to receive the highest accuracy on the CIFAR-10 data set. Shallow networks could never reach the accuracy of the deep networks, even though we trained them by a great method, distillation. Student models learned the complex characteristics of the teacher networks and improved their accuracy significantly. Nevertheless, they performed worse than the deep convolutional networks. We extended the work presented in [1] by including skip-networks with residual networks. These networks were in complete agreement with the results presented in [1].

1 Introduction

Do deep convolutional nets really need to be deep and convolutional? Nowadays it is common knowledge that a network with a large enough single hidden layer of sigmoid units can approximate any decision boundary . However, empirical work shows that it is difficult to train shallow networks to be as accurate as deep networks in practice. This fact can clearly be seen in [1].

In this paper, we will compare deep convolutional and linear networks to their shallow counterparts. To keep the playing field even we chose to have a constant number of parameters for all scenarios. We will train the shallow networks using distillation training as done by [1]. We will also improve upon the teacher and student models used in [1] by using ResNets [2].

The importance of our paper lies in the fact that we are challenging the assertions made by [1]. If in fact, our results differ from [1], the question of if indeed deep convolutional networks are needed to acquire good performance for computer vision will be answered.

We found that the assertions made by [1] are consistent with our experiments. Our experiments cleary shows that a deep convolutional network is needed to obtain good performance in computer vision. Our experiments also show that convolutional networks performs far better than deep and shallow linear feed forward networks and that the best performer is by far deep convolutional networks.

1.1 Hypothesis

The model has to be deep and it has to be convolutional in order to obtain a great accuracy on the CIFAR-10 data set.

2 Related work

The inspiration for this project comes from the paper [1], which in turn is and extension of [3]. [1] performs a thorough set of experiments challenging the question "Do deep convolutional nets really need to be deep and convolutional?". This paper presents several significant results that strengthen the hypothesis that depth of the network and the network being convolutional are crucial characteristics of

Preprint. Under review.

the network when it comes to accuracy on the test data. This has been done by using state-of-the-art techniques such as distillation training and Bayesian optimization.

In our work, experiments are performed on the same data set, CIFAR-10. Nevertheless, the focus of our work is slightly extended towards skip connections, and less time is spent on the optimization methods.

3 Data & Data Augmentation

The dataset that was used is the CIFAR-10 set [4]. The dataset contains 60000 images with 10 categories which are: dog, frog, horse, ship, bird, cat, deer, truck, airplane, automobile. Each image is the size 32x32 and with three color channels. The dataset will be partitioned into 40 000 samples for training, 10 000 validation, and 10 000 test sets. In addition, the dataset for training will be normalized with the help of the standard deviation and the mean. In order to increase the ability of generalization for our network, we performed the data augmentation by extending the data with random horizontal flips and random crops.

4 Methods

In order to recreate some of the results presented in [1]. Google Cloud platform was used, where all four group members could run PyTorch code on a Nvidia K80 GPU. Furthermore, since our primary goal was not on providing optimal accuracy of the models, Adam optimization in PyTorch was used instead of the Bayesian optimization.

Note that in all models in this project we used batch size of 32 and we performed batch-normalization.

4.1 Convolutional neural networks (CNN)

As in [4] convolutional neural networks, CNN's, are utilized to investigate performance. Moreover, the approach is to develop two models of the CNN's to allow for comparison of the performance between the CNN's and the other proposed models. The architecture for the CNNs that was used in the experiments can be found in section 4.3.

4.2 Residual Networks (ResNet)

A proposed solution to the problem of degradation in a deep network architecture is using skip connections as done by [2]. We use skip connections to construct networks that give a state-of-the-art performance to use as teacher networks.

The architecture we use is a skip connection between a stack of 2 convolutional layers as done by [2].

4.3 Teacher and student models

4.3.1 Teacher models

Since [1] is our main inspiration for the project, we use one of its best performing teacher models in order to train our student networks. The architecture of this teacher model that we call Teacher-1 can be observed in the first row of Table 1. Note that we implemented four drop outs in the Teacher-1 model, drop out with probability 0.2 after first max pool, probability 0.3 after second max pool, probability 0.37 after third max pool, and probability 0.42 after second fully connected layer. Furthermore, a deep convolutional ResNet teacher model was used with architecture that can be observed in second row in the Table 1.

Where c stands for convolutional layer, mp for max pooling, fc for fully connected layer. The integer number before each layer stands for the number of nodes and the exponential for the number of such adjacent layers. The convolutional filters used (in all of the models with such layers in this project) are all of the size 3×3 with padding of one and stride of one.

Model	Architecture	Parameters
Teacher-1	$76c^2$-mp-$126c^2$-mp-$148c^4$-mp-$1200fc^2$-$10fc$	5M
Deep ResNet	$64c^8$-$128c^{12}$-$256c^{16}$	10M

Table 1: Table of network architecture for teacher models.

4.3.2 Student models

The five student models that was used in this project with the following names and architectures can be observed in the Table 2.

Model	Architecture	Parameters
3-layer-1-CONV	$64c$-mp-$640fc$-$10fc$	10M
4-layer-2-CONV	$64c^2$-mp-$600fc$-$10fc$	10M
3-layers, all linear,	$2048fc^2$-$10fc$	10M
4-layers, all linear,	$1200fc$-$2400fc$-$1800fc$-$10fc$	10M
Shallow-ResNet	$128c^2$-$512c^4$-$10fc$	10M

Table 2: Table of network architecture for student models.

4.4 Deep linear network

A deep linear network with 20 layers and approximately 10M parameters was created in order to compare it with the two other deep models. In order to strengthen our hypothesis, that the best performing models have to be deep and convolutional, this model ought to have lower accuracy than the two deep convolutional models. The models architecture: $256fc^8$-$512fc^6$-$1024fc^8$

4.5 Model Compression and Distillation

In order to strengthen the first part of our hypothesis, that the best performing models have to indeed be deep, we should show that training several shallow models, trained by a great training technique, still do not yield the same performance as the deep models.

One such technique is model compression. It is based on training the student models with the output of the teacher model instead of the data label. The idea is that the student will directly learn the characteristics of the deep network, hence obtain higher accuracy.

4.6 Activation and Loss

For all models we use a ReLu activation function at each layer. When training on the CIFAR-10 dataset we use a Cross entropy loss. When we perform distillation training we use MSE loss where we have applied softmax to each output vector from the student and teacher. Note that [1] uses logits but in doing so we found worse performance and therefore applied softmax to these logits before evaluating the loss.

5 Experiments

Our experiments were performed on the CIFAR-10 data and are presented in the following tables. Table 3 compares distillation training results vs training the very same models on labels provided by the data set. Table 4 is an extension of the results of some of the previous models trained with the deep ResNet model as a teacher. Table 5 contains the performance of the Teacher-1 model and Deep-ResNet. Table 6 contains the performance of the deep linear model.

Please note that all the presented number of parameters are always either rounded up or down, the numbers are approximate.

5.1 Distillation training results

The following tables have five columns each, the first column labeled "Model" indicates what model was used. The second column is the number of parameters in the model. The third column is the accuracy of this model without using any distillation method. The fourth column is the improved accuracy after distillation. The final fifth column shows the number of epochs without distillation vs with distillation. Note that since training with distillation did not overfit as fast as without distillation, more epochs were run.

Model	Number of parameters	Accuracy-no-distillation	Accuracy-distillation	Epochs
3-layer-1-CONV	10M	74.27%	75.57%	75 vs 300
4-layer-2-CONV	10M	79.04%	80.62%	75 vs 300
3-layer-linear	10M	55.12%	58.09%	75 vs 300
4-layer-linear	10M	56.87%	58.39%	75 vs 300
Shallow-ResNet	10M	88.72%	89.63%	75 vs 200

Table 3: Model accuracies with and without distillation using Teacher-1 as teacher on CIFAR-10.

Model	Number of parameters	Accuracy-no-distillation	Accuracy-distillation	Epochs
3-layer-1-CONV	10M	74.27%	76.01%	75 vs 300
4-layer-2-CONV	10M	79.04%	80.96%	75 vs 300
3-layer-linear	10M	55.12%	58.37%	75 vs 300
4-layer-linear	10M	56.87%	59.22%	75 vs 300
Shallow-ResNet	10M	88.72%	89.87%	75 vs 200

Table 4: Model accuracies with and without distillation using Deep-ResNet as teacher on CIFAR-10.

Note that the presented results for the distillation trained student models are restricted by 300 epochs, due to the time constraint that we had for this project.

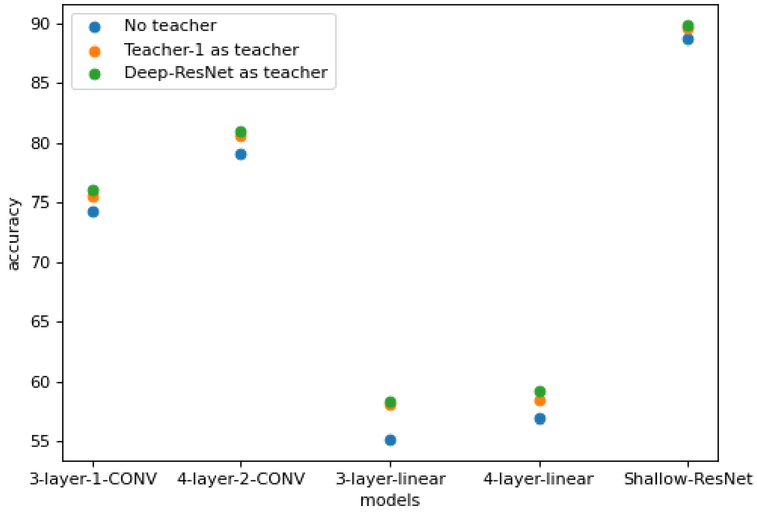

Figure 1: Accuracies when training without distillation (no teacher) and with distillation for two different teachers for all the different student models. (This plot illustrates the results already presented in table 3 and 4)

5.2 Teacher models

In order to see the performance of the teacher models on CIFAR-10 look at table 5.

Model	Number of parameters	Accuracy	Epochs
Teacher-1	5M	90.49%	250
Deep-ResNet	10M	91.54%	250

Table 5: Teacher models performance on CIFAR-10.

5.3 Deep linear model

In order to see the performance of the deep linear model on CIFAR-10 look at table 6.

Model	Number of parameters	Accuracy	Epochs
Deep-linear	10M	51.15%	75

Table 6: Deep linear network performance on CIFAR-10.

5.4 Loss and accuracy plots

Instead of presenting many loss and accuracy plots, we show four plots for the 3-linear-model, since all of the other plots had the very same behaviour.

Firstly in figure 2, loss plots are shown where the left plot is the validation and training loss after using distillation with Teacher-1. As can be seen, the validation loss is still decreasing after the 300 epochs, which is not the case for the right plot where we have the very same model but without distillation training. The validation loss start to increase quickly. This can be compared with figure 3, that present their accuracy. As soon as the validation loss starts to increase in the right plot of figure 2, the accuracy of the model stop increasing. When we use distillation, we obtain a very different result, the accuracy of the model that can be seen in the left plot of figure 3 is always increasing during the 300 epochs and we have seen that the validation loss is always decreasing for these epochs.

It is important to note that the validation loss could be decreasing for more epochs than we trained our models and that the accuracy for the model using distillation could be slightly higher after more epochs run.

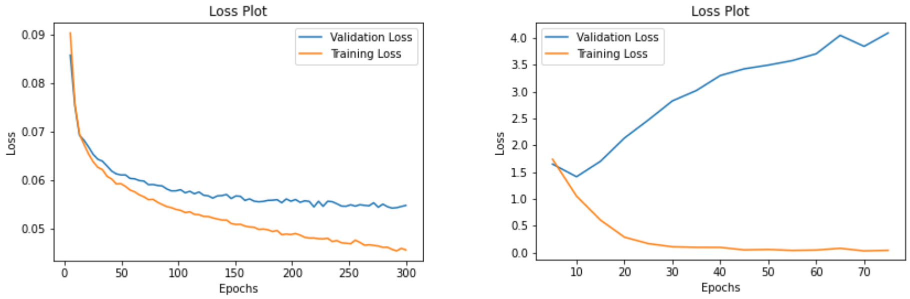

Figure 2: Loss plots with (left) and without (right) distillation for 3-layer-linear model.

6 Conclusions

All of our results indicate that the hypothesis can not be rejected. Deep convolutional networks provide the very best accuracy on the CIFAR-10 data set. Deep linear models perform very poorly, and no shallow models can obtain the accuracy of the deep ones, even after distillation training. Thus, our results are strengthening our hypothesis. Moreover, applying the ResNet technique did improve our results on both the techer convolutional model and student convolutional model in distillation.

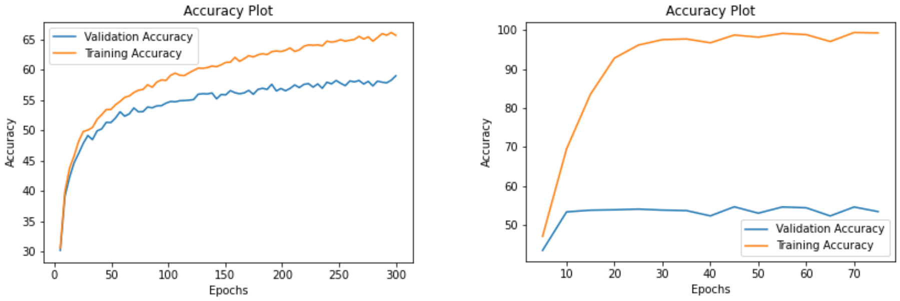

Figure 3: Accuracy plots with (left) and without (right) distillation for 3-layer-linear model.

Another interesting result was the results of distillation training, which gave higher accuracy for all the student models. In addition, the presented plots of validation loss show that the loss is decreasing for a greater number of epochs, which is not the case when training on the labeled data, where the network tends to converge to its limiting accuracy quite fast. Therefore, the accuracy that we obtained and presented on some of the student models is not optimal and could be slightly increased by training for a greater number of epochs. This can be seen in the accuracy plot of figure 3 where the validation accuracy has yet to converge. Unfortunately, due to time and resource constraints we could not run the training for a longer period of time

References

[1] Gregor Urban, Krzysztof J Geras, Samira Ebrahimi Kahou, Ozlem Aslan, Shengjie Wang, Rich Caruana, Abdelrahman Mohamed, Matthai Philipose, and Matt Richardson. Do deep convolutional nets really need to be deep and convolutional? *arXiv preprint arXiv:1603.05691*, 2016.

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[3] Jimmy Ba and Rich Caruana. Do deep nets really need to be deep? In *NIPS*, 2014.

[4] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.