Compression techniques for Semantic Segmentation

Jatin Palchuri
MS, Computer Engineering,
New York University, Tandon School Of Engineering

Tamoghna Chakraborty
MS, Computer Engineering,
New York University, Tandon School Of Engineering

Abstract

Deep learning models are very computationally expensive because of their overparameterization; they require large memory space and the number of computations increases per layer of a neural network. To run such computationally heavy models on hardware limited devices is a big challenge. The common techniques to compress a neural network are weight pruning, quantization, knowledge distillation, etc. In this project, we are looking at compression techniques for semantic segmentation networks.

1 Introduction

Deep convolutional Neural Networks provide high accuracy in semantic segmentation. One of the main problems with such networks is that they have a huge set of trainable parameters. Implementing these models on embedded systems or mobile devices (hardware restricted environments) requires an efficient way to compress these models ("efficient": after compression, the model has to have similar accuracy to the original model).

There are well known deep compression techniques like pruning, trained quantization, Huffman coding, knowledge distillation for many image classification tasks. From [4],[3] we can clearly see that only pruning or only quantization leads to erroneous models, so future works combined these techniques with knowledge distillation which resulted in better models as we know from [8].

Recently, developing compact networks for segmentation applications has become very popular because of the high demand in mobile applications. Our focus is on the training of such networks with the help of traditional (large) networks, and try improving segmentation accuracy.

Pixelwise distillation is an approach we adopted where we apply knowledge distillation at the pixel level, essentially breaking down the segmentation problem into several pixel classification problems.

Pairwise similarities between pixels from both networks are aligned using pairwise distillation process which helps preserve structure of model.

Lastly, we use a process called quantization aware Knowledge Distillation in order to combine knowledge distillation with quantization as their combination is very promising in efficient compression. This is similar to the work done in [6]

35th Conference on Neural Information Processing Systems (NeurIPS 2021), Sydney, Australia.

2 Related work

2.1 Semantic Segmentation

For carrying out semantic segmentation, the best solution is the use of deep neural network models. But, for great accuracy results, these models are very large and cumbersome. These models sacrifice speed and efficiency for higher accuracy. Hence, a lot of work has been done in order to increase speed and efficiency of these networks so that they could also be used for mobile applications. MobileNet [5] is an example of such work.

2.2 Knowledge Distillation

As the name suggests, knowledge distillation is a technique used to distill knowledge from a teacher network to a student network. In the context of compression, the teacher network is larger than the student network. There are several approaches that others have taken for this knowledge transfer and many of them are listed by [7], e.g. using the class probabilities of the larger model as soft targets for training the compressed model, etc. For knowledge distillation of complex tasks like object classification, there has been some work on Hint-based learning [2] for student networks which reduce loss on both bounding boxes and the hints received from the teacher network.

2.3 Quantization

This is a process of reducing the number of bits required in representing information. In neural networks, quantization is the process of representing the weights and activations of a layer in lower bit counts. This is done by using a quantization function which scales the given weight vector into a quantized form. The quantization functions used in our work is explained in detail here [9]

3 Methodology

In this work, we start by using a U-NET [1] as a student model and train this network for semantic segmentation task, then we quantize this network to get the quantized model. After this, we perform knowledge distillation on the pretrained student model using a large pretrained teacher network trained on semantic segmentation from PyTorch (RESNET101-DeepLabV3).

3.1 Structured knowledge distillation for semantic segmentation

Since in semantic segmentation, we have to label each pixel to a class, the knowledge distillation can be performed by looking at each pixel's softmax output or by comparing different local pixel areas between teacher and student. Finally, we tried an approach where the student network is quantized while knowledge distillation is in progress. [7]

Pixelwise Distillation: Since semantic segmentation is a pixel labelling problem, we can transfer the knowledge from teacher to student networks using each pixel's softmax probabilities and comparing the KL-divergence between all the pixels and using this as a loss function to minimize.

Pairwise Distillation: Since an image has pixelwise similarities too, to capture this information from the teacher to the student, first we compute the similarity between the features of the local pixels and then compare these similarities produced by both teacher and student using a squared difference.
The main objective of the student network is to reduce the pixelwise and pairwise distillation loss along with the classification loss.

3.2 Quantization aware knowledge distillation

We have a 'fake' quantization module that is present during knowledge distillation: all the computations are still made using higher precision, but the grouping of weights is done during training.

4 Results

We use PASCAL VOC2012 segmentation dataset which contains around 1440 images for each train and test.
For training the U-NET model from scratch, we used Adam optimizer with a learning rate of 10^{-4} and a momentum of 0.8 along with a learning rate scheduler operating in min mode. After running for 1000 epochs, the loss saturates at 50.

We suspect that since the model is not converging to a local minima, knowledge distillation is not working well because most of the knowledge distillation techniques available to us use a student network trained on some number of epochs which tells us that the student has to be in the direction of local minima

Figure 1 shows that after 1000 epochs, we run for 20 more epochs just to show that the loss is saturated

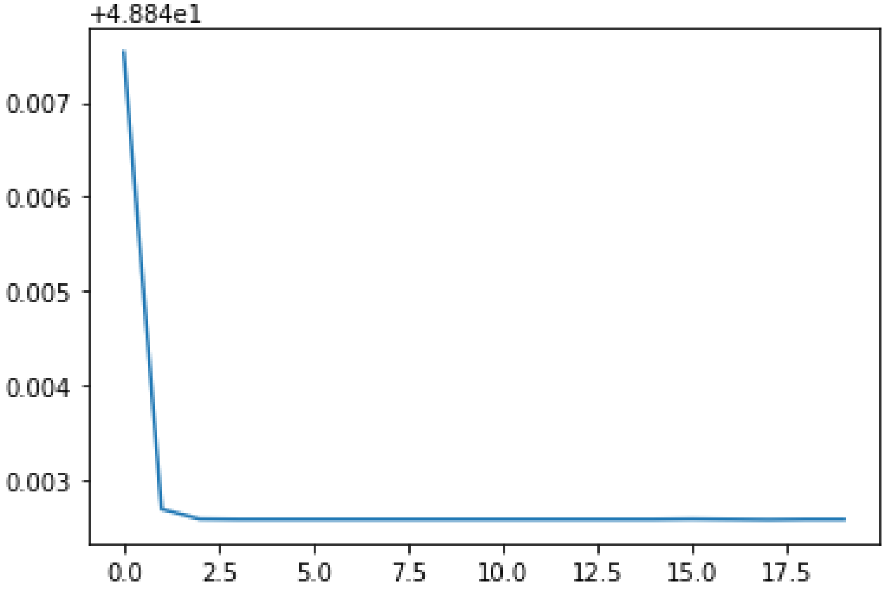

Figure 1: U-NET training

Figure 2 shows that knowledge distillation is not being performed as expected.

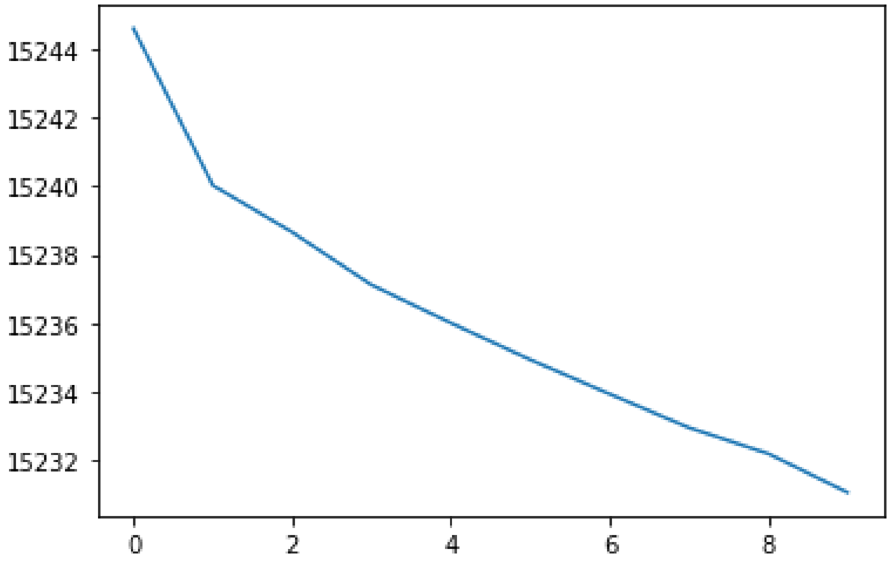

Figure 2: Knowledge Distillation

References

[1] M. Buda, A. Saha, and M. A. Mazurowski. Association of genomic subtypes of lower-grade gliomas with shape features automatically extracted by a deep learning algorithm. *Computers in biology and medicine*, 109:218–225, 2019.

[2] G. Chen, W. Choi, X. Yu, T. Han, and M. Chandraker. Learning efficient object detection models with knowledge distillation. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 742–751, 2017.

[3] Y. Cheng, D. Wang, P. Zhou, and T. Zhang. A survey of model compression and acceleration for deep neural networks. *arXiv preprint arXiv:1710.09282*, 2017.

[4] Y. Choi, M. El-Khamy, and J. Lee. Towards the limit of network quantization. *arXiv preprint arXiv:1612.01543*, 2016.

[5] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[6] J. Kim, Y. Bhalgat, J. Lee, C. Patel, and N. Kwak. Qkd: Quantization-aware knowledge distillation. *arXiv preprint arXiv:1911.12491*, 2019.

[7] Y. Liu, K. Chen, C. Liu, Z. Qin, Z. Luo, and J. Wang. Structured knowledge distillation for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2604–2613, 2019.

[8] I. Oguntola, S. Olubeko, and C. Sweeney. Slimnets: An exploration of deep model compression and acceleration. In *2018 IEEE High Performance extreme Computing Conference (HPEC)*, pages 1–6. IEEE, 2018.

[9] A. Polino, R. Pascanu, and D. Alistarh. Model compression via distillation and quantization. *arXiv preprint arXiv:1802.05668*, 2018.