Optimizing Transfer Learning: A Shrinking Approach for Enhanced Few-Shot Image Classification

Prakhar Gupta[†], Jay Jawale[†], Vibesh Kumar[†]

[1]SMCS, IIT Goa,.

Contributing authors: prakhar.gupta.21031@iitgoa.ac.in; jay.jawale.21063@iitgoa.ac.in; vibesh.kumar.21031@iitgoa.ac.in;
[†]These authors contributed equally to this work.

Abstract

Transfer learning has revolutionized image classification by enabling the adaptation of pre-trained models to specific tasks with limited datasets. In this study, we propose a methodology for shrinking deep Convolutional Neural Networks (CNNs), in particular, ResNet architectures, to enhance computational efficiency and accuracy, especially during few-shot learning. By removing intermediate layers, we preserve larger models' low-level feature extraction capabilities while significantly reducing their size and complexity. Our experiments show a significant improvement from the traditional ResNet-50 and ResNet-18 models during few-shot training. This paper suggests a new direction for further exploration into optimizing deep learning architectures for transfer learning, extending its applicability to other pre-trained models.

Keywords: Deep Learning, Transfer Learning, Image Classification, ResNet, Few-Shot Learning, Computer Vision

1 Introduction

Image classification is among the most important applications of Deep learning mechanisms in Computer Vision. Various Models have been proposed to make this task possible. Deep CNNs have achieved great success in the tasks related to image classification. Large models pre-trained on massive labelled image datasets have made the task of Image classification much faster and more efficient. Transfer learning [1] has been primarily adopted to use these pre-trained models for specific tasks. Transfer learning is a technique where a model is trained for a particular task and re-used for another related task. This allows training models even in the absence of a large dataset.

The lower levels of a pre-trained model generally extract the low-level features of the input while the upper layers primarily extract more high level, complex and specific features from the dataset [2]. The transferability of the low-level features is more than the high-level features. To exploit this, most approaches to transfer learning involve fine-tuning the pre-trained model on a smaller "target" dataset by unlocking the later layers. Thus, the resultant model can make better predictions on the new dataset despite its limited size owing to its earlier layers being trained on a larger dataset.

This approach allows efficient training of a model on a small dataset. Zhi et al. [3] explores an alternate approach to deleting the upper layers from a trained Deep CNN. In this paper, we explore the effects of deleting intermediate layers to not change the core architecture of the original model. We propose a methodology similar to that, along with transferring information from a larger trained model, also reduces the model size and allows faster computation.

2 Models and Dataset

We make use of the ResNet models [4] as our CNN models for image classification. ResNet, standing for Residual Networks, makes use of residual connections or skip links to allow gradients to flow directly to earlier layers. This solves the problem of vanishing gradient allowing for much deeper networks and achieving state-of-the-art performance on image classification datasets. In this paper, we make use of the ResNet-50 and ResNet-18 models which consist of 50 and 18 convolution layers respectively.

The ResNet-50 architecture (Figure 1) consists of a convolution layer and a max pool layer, followed by 4 bottlenecks, with each bottleneck consisting of multiple blocks. Each block consists of 3 convolution layers. The first bottleneck has 3 blocks, the second 4, the third 6, and the 4th has 3. This gives a total of 50 layers.

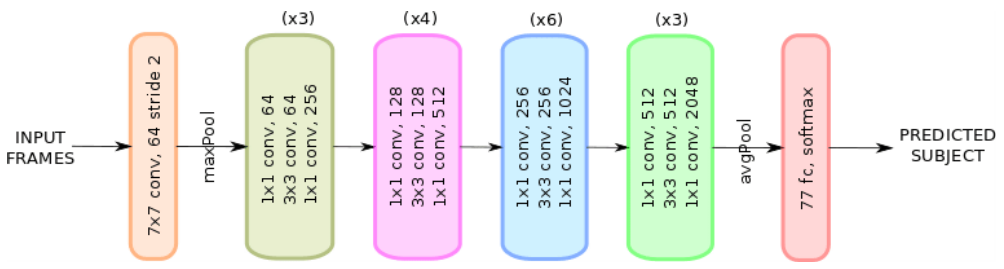

Fig. 1 ResNet-50 architecture [5]

To train the ResNet models, we make use of the CIFAR-10 and the CIFAR-100 datasets [6]. The CIFAR-10 dataset consists of 10 classes, with each class containing 6,000 32x32 images for a total size of 60000 images. Similarly, the CIFAR-100 dataset consists of 100 classes with 600 32x32 images.

The datasets are very limited considering the small number of samples and the poor resolution of the images. In order to preserve the baseline statistics, we make no attempts to augment the dataset and train our models on the original dataset. We make a split of the dataset into training set with 50000 images and a validation set containing 10000 images.

3 Methodology and Experiments

3.1 Methodology

We first train a deeper model (in our case ResNet-50 model) on a large dataset until convergence. This allows the model to learn the base-level features from the images in the dataset. We delete a certain number of end blocks from each bottleneck, ensuring that each bottleneck has blocks that can hold the lowest-level features.

This new model is then re-trained on the images from the target dataset. We experiment with several methods of re-training the model on the target dataset. Firstly, we retrain the model in its entirety on the new dataset for, both, a small number of epochs and then until convergence. Then, we freeze the initial layers of the model and only unlock the final layers for a faster fine-tuning during few-shot training on the new dataset.

3.2 Experiments

As the larger model, we consider the ResNet-50 model. We train the ResNet-50 model on the CIFAR-10 dataset until convergence. Then, we shrink the model until it approximately matches the size of the ResNet-18 model. From the first bottleneck, we delete the last block, keeping the first 2. In the subsequent bottlenecks, we only keep the first block and discard the rest. As a result, we have a new model with 17 convolution layers.

We train this shrunken model on the CIFAR-100 dataset in the various ways mentioned in Section 3. We make comparisons between each of these models and also with a fully trained ResNet-50 model and a fully trained and fresh ResNet-18 model. It is important to note that the ResNet-50 model is significantly larger (23.9M parameters) than the ResNet-18 model (11.4M parameters). We test the accuracy of the classifications on the validation sets and plot the accuracy across several epochs. For Few-shot learning we train the model for 3 epochs while for a fully trained model we train the model until 50 epochs in order to save computational time.

4 Results

In models trained fully until 50 epochs (Figure 2 and Figure 3), the new model achieved a slightly better accuracy of over the traditionally trained ResNet-18 model with an accuracy of . Both these models achieved a significantly superior performance than the ResNet-50 model.

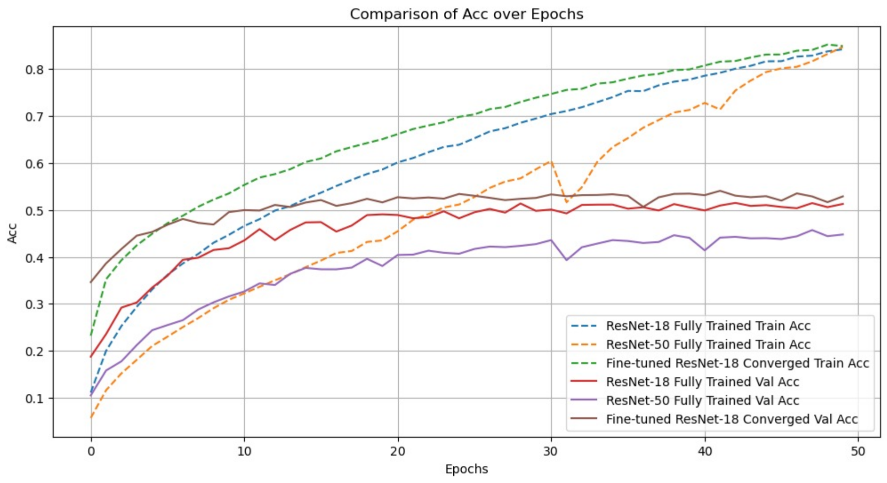

Fig. 2 Accuracy vs Epoch-50

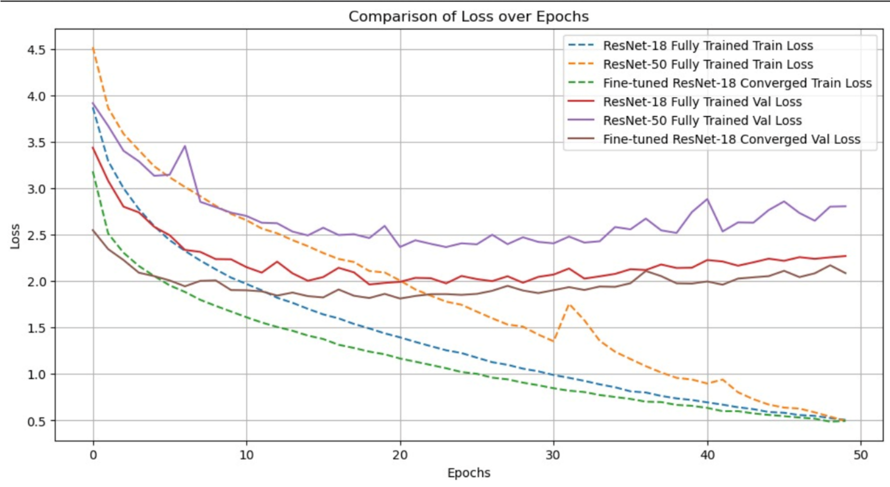

Fig. 3 Loss vs Epoch-50

During few shot training (Figure 4 and Figure 5), the proposed model performed significantly better than the fresh ResNet-18 model. The model trained with initial layers frozen showed a slightly faster improvement in accuracy.

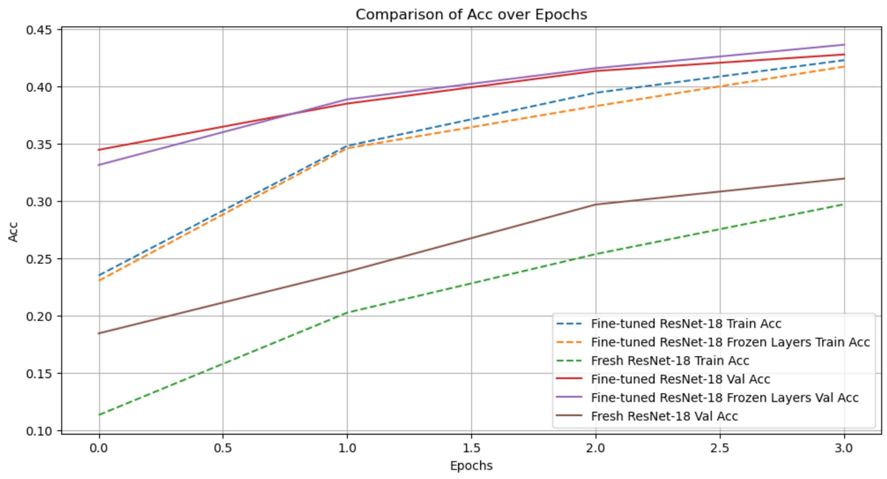

Fig. 4 Accuracy vs Epoch- Few-Shot

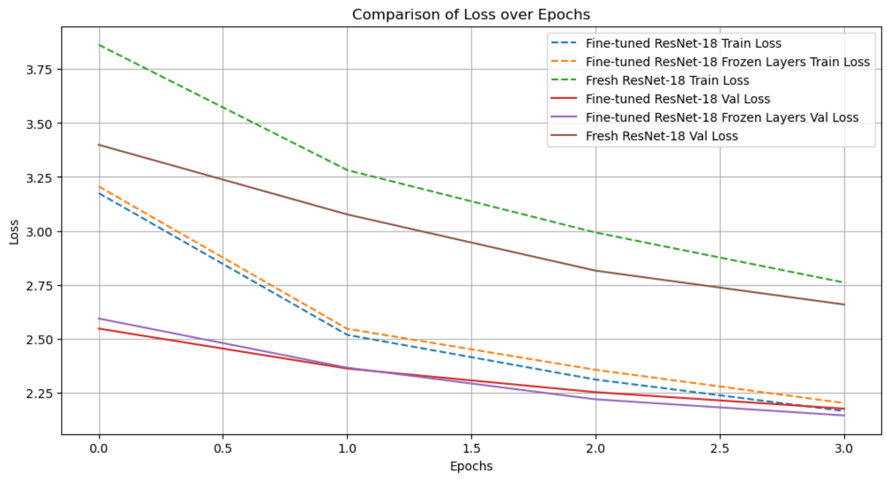

Fig. 5 Loss vs Epoch- Few-Shot

Model	Few shot	Fully trained
ResNet-50	-	45%
ResNet-18	31.93%	51.49%
Proposed Model - Unfrozen	42.75%	**54.07%**
Proposed Model - Frozen	**43.60%**	-

Table 1 Best Validation accuracy on CIFAR-100

5 Discussion and Conclusion

The new proposed model can outperform the traditional ResNet-18 and ResNet-50 models in both longer training times and few-shot learning. This difference is much more significant in few-shot learning with an improvement of almost 12% from a fresh ResNet-18 model. The proposed model also outperforms the ResNet-50 model and is significantly smaller in size. This significant improvement can help us train smaller models that have a faster computation incase of limited computational resources.

This can be attributed to the fact that the new model has a higher accuracy from the low-level features being extracted from the original model. The much more efficient training in few-shots is significant when combined with the much smaller size than the original large pre-trained model.

The poor performance of ResNet-50 as compared with ResNet-18 model can be attributed to the difficulty in training of a deeper network and a small dataset.

6 Future Direction

This paper presents a possible direction of research that can result in smaller models with significantly better few shot training. Further research in shrinking larger models and layer deletion is needed to find the optimum model that can be achieved with the proposed methodology. This paper has worked on ResNet model but the proposed methodology holds the potential to have significant results on other large models such as DenseNet and EfficientNet. Moreover, deleting layers from a much better pre-trained model, such as a ResNet-50 model trained on the ImageNet dataset, can significantly improve the few-shot learning for a shrunken model.

References

[1] Hussain, M., Bird, J., Faria, D.: A study on cnn transfer learning for image classification. (2018)

[2] Yosinski, J., Clune, J., Bengio, Y., Lipson, H.: How transferable are features in deep neural networks? (2014). https://arxiv.org/abs/1411.1792

[3] Zhi, W., Chen, Z., Yueng, H., Lu, Z., Zandavi, S.M., Chung, V.: Layer removal for transfer learning with deep convolutional neural networks, pp. 460–469 (2017). https://doi.org/10.1007/978-3-319-70096-0_48

[4] He, K., Zhang, X., Ren, S., Sun, J.: Deep Residual Learning for Image Recognition (2015). https://arxiv.org/abs/1512.03385

[5] Jahromi, M., Buch-Cardona, P., Avots, E., Nasrollahi, K., Escalera, S., Moeslund, T., Anbarjafari, G.: Privacy-constrained biometric system for non-cooperative users. Entropy **21**, 1033 (2019) https://doi.org/10.3390/e21111033

[6] Krizhevsky, A.: Learning multiple layers of features from tiny images, 32–33 (2009)