Differentiating between presence or absence of contrast agent and contrast mechanisms on MR images

Neville D Gai
ngai1@stanford.edu
SUNet ID: 06605751

Mentor: Shubhang Desai
Category: Healthcare

Purpose

Identifying MR images as pre- or post-contrast can be valuable in practice. This can have implications for archive management and image retrieval from large databases at different sites and across different scanners. Another purpose could be as an automated teaching tool for radiologists and technicians. Image segmentation can use additional information from different contrast mechanisms to improve intensity-based segmentation and help separate class distributions. Dicom headers (if available) are not always reliable and could be corrupted. In addition, nomenclature used for images from sequences can vary considerably across scanners and sites. Manual sorting of images based on contrast characteristics can be tedious and impractical. Additional categorization of MR image types was also considered.

Introduction

Convolution neural networks (CNNs) excel in classification of images where layers of filters successively extract relevant features. Several deep CNNs have been developed and tested on large data sets such as ImageNet categorizing more than a million images into a 1000 different classes. Some of the popular such CNNs include AlexNet [1], VGG [2], Inception [3], ResNet [4] and Xception [5]. Training such deep networks from scratch requires extremely large data sets, considerable memory requirements and computation time as well as high-end processing performance (GPUs). However, once trained, such networks can be repurposed for a different application provided the tasks bear some similarity.

A couple of earlier works [6, 7] tried to address the MR image classification problem. Differences with earlier works and the work presented here are highlighted in Discussion.

Transfer learning provides the advantages of a better initial model, higher learning rate for similar problems, higher accuracy after training and faster convergence to a desired performance level. In this project, transfer learning was used for the task of identifying MRI images based on their contrast characteristics.

Methods

Data
Our laboratory has a dataset consisting of thousands of anonymized Dicom images with different contrast characteristics based on MRI sequence and presence or absence of contrast agent. The first task was to curate these images to make them suitable as input to a CNN. T1-weighted and T2-weighted pre- and post-contrast images were considered. Grayscale Dicom images from these classes were first converted to single-channel JPEG images in Matlab® (code available on request). Slices with minimal brain tissue were excluded from the training and testing datasets. Images were binned into their correctly labeled sub-directory. A total of 2894 images corresponding to the 4 classes was used for training, validation, and testing. Training and testing images were from different subject studies. See Table 1 for distribution across

classes for training and testing. Note the balanced data sets. Sample T2-weighted FLAIR (FLuid Attenuated Inversion Recovery) images from the training and testing sets are shown in Appendix A.

	T1-w pre	T1-w post	T2-w pre	T2-w (FLAIR) post
Training and validation set	629	629	632	619
Testing set	97	97	97	94

Table 1: Image distribution across the 4 classes for training and testing.

Model

Two models pre-trained on ImageNet data set were considered for this study based on work by Ke et al. [8] which showed superior performance by ResNet and Inception CNNs, pretrained on ImageNet, for medical imaging tasks. In particular, ResNet50 and InceptionV3 were studied as potential candidates due to their intermediate complexity and superior performance. The top layer of the model was modified for the purpose of classifying into four categories. Accordingly, the final convolutional layer of the pre-trained model was connected to a *Dense* fully connected layer with 1024 neurons which was then connected to a *Dropout* layer to achieve some regularization. Finally, a Dense layer with *softmax* activation was used to classify the images. Grayscale input images of size (224, 224, 1) were fed to the network after convolution with a filter to expand channel size from 1 to 3. Schematic model is shown in Appendix B, Figure 1.

Fine Tuning

Four different layer configurations were considered for the pre-trained part of the network: (a) all frozen (b) all trainable (c) first 125 of 175 layers frozen and (d) first 150 layers frozen.

Hyperparameter selection

For layers which could be regularized in the base model, L2 regularization was set with value 0.0001. *Adam* optimizer with default values (lr = 0.001, $\beta_1 = 0.9$, $\beta_2 = 0.99$, $\varepsilon = 10^{-7}$) was used. Different learning rates were considered: 0.01, 0.001 and adaptive rate. Mini-batch size was initially set at 16 but later changed to 32 to reduce fluctuations in loss/accuracy with the early stopping model.

Training

The modified CNN model was implemented in Keras with TensorFlow 2.0 locally before uploading to Google Colab. Image data were zipped and uploaded to Google Drive before unzipping to Colab. Colab used TensorFlow version 2.4.1 or 2.5. Since differentiation was between four classes, *categorical crossentropy* was used as the loss function. Shuffling was employed with 15% of the data assigned for validation. Image normalization was not done since batch normalization follows convolution in the ResNet architecture. Early stopping with varying min(Δval_acc) and *patience* was employed to get the best results for model comparison and training in the shortest possible time. Table 2 provides details about the various hyperparameters and models engaged for the task.

Data augmentation was not considered for two reasons. Flipping images is not a good idea for this application, since the brain is not symmetrical about the longitudinal fissure. Secondly, our trained model did exceedingly well on validation and testing, thereby obviating the need for data augmentation.

Schedule	Learning Rate	Minimum Δ	Patience	Frozen Layers	Base Model	Epochs	Accuracy (%)
1	0.001	0.0002	3	All	ResNet 50	10	99.21
2	0.001	0.0002	3	All	Inception V3	16	90.03
3	0.001	0.0002	3	None	ResNet 50	9	80.05
4	0.001	0.0002	3	125	ResNet 50	14	99.21
5	0.001	0.0002	3	150	ResNet 50	9	98.95
6	0.001	0.0001	5	All	ResNet 50	11	98.43
7	0.001	0.0001	5	150	ResNet 50	7	99.48
8*	0.001	0.00001	7	150	ResNet 50	24	99.74
9	0.01	0.00001	7	150	ResNet 50	16	99.21
10	Adaptive	0.00001	7	150	ResNet 50	11	95.28

Table 2: Training and testing were carried out for various values of hyperparameters and models to ascertain the best combination for the current task. Training stopped when no improvement of at least Δ in validation accuracy was registered over the last several epochs (corresponding to *patience*). Schedule 8 provided the best result.

Testing

The model was tested on a total of 385 images obtained from studies that were similar to, but mutually exclusive of, the training and validation set. Only accuracy (# of images correctly classified/total # of images) was measured because of the excellent performance achieved as described in Results.

Model Analysis

The final model was analyzed to obtain class activation maps (CAM). Heat maps were obtained by modifying the training model to include a global average pooling layer after the last convolution layer of base ResNet50 and prior to the Dense *softmax* layer. This modified model was trained using schedule 8. Figure 2 in Appendix B shows the schematic for the modified model.

Results

Training and Model Selection

Table 2 shows the training schedule and testing results. Training took approximately 22 s to 45 s per epoch based on number of trainable layers and available GPU (NVIDIA® Tesla T4 or K80). Based on comparison of number of epochs and testing accuracy, ResNet50 was selected over InceptionV3. Fine tuning on the number of frozen layers indicated superior performance with first 150 layers frozen (up to conv5_block1_3) and with all layers frozen over all layers trainable and trainable after 125 layers (after conv4_block5_1). On further extending the stopping criterion, the model with first 150 layers frozen provided the best performance. (See Appendix C for detailed analysis and plots.)

Testing

Testing accuracy for various schedules is shown in Table 2. The highest accuracy achieved was 99.74%, which corresponded to just 1 misclassified image of 385 test images. A post-contrast T1-weighted image was misclassified as a pre-contrast T1-weighted image.

Model Analysis

Examples of sample images with and without an overlaid heat map (including the one image which was misclassified) are shown in Figures 1 and 2 for all four classes. Pre- and post-contrast images typically differ visually due to presence of enhanced signal in vascular spaces such as vessels and in vascularized tumors. Visual inspection of the heat map reveals that the model was classifying by focusing on areas of the brain which exhibited differences. For example, pair of pre- and post-T1 weighted images were differentiated based on the area near the sagittal sinus (red arrow) which shows hyperintensity. Pre-and post- T2 weighted images were differentiated based on hyperintensity resulting from soft tissue trauma and presence of cerebrospinal fluid (CSF) in the sulci (orange and yellow arrows). In this case, the FLAIR T2-w images differ from pre-contrast T2-w images mainly based on these two characteristics rather than contrast agent-based changes which are more subtle.

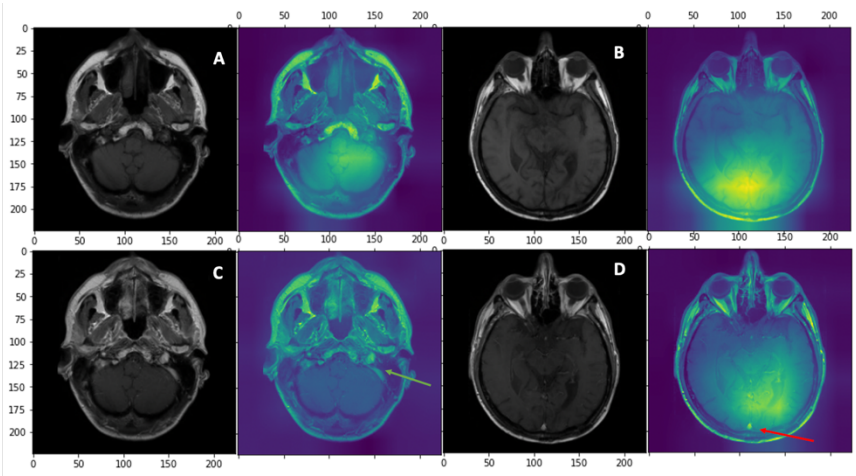


Figure 1: Pre- (A and B) and corresponding post-contrast (C and D) T1-weighted images alongside their overlaid activation maps. Red arrow indicates the location which enhanced due to presence of vessel. Image C was the only misclassified image (of 385 images) as the heat map failed to localize on the area of vessel enhancement (green arrow).

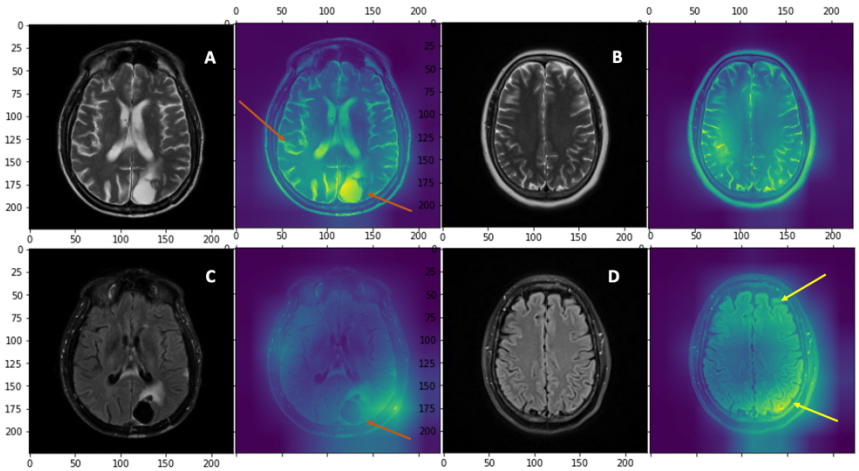


Figure 2: Pre- (A and B) and corresponding post-contrast (C and D) T2-weighted images and their overlaid heat maps. Orange arrows indicate hyperenhancement in pre-contrast images from soft tissue trauma (hypointensity in post-contrast T2-w FLAIR) and from CSF. Yellow arrows indicate locations of CSF absent in T2-w post-contrast FLAIR images.

Discussion

Excellent results were achieved after fine tuning the selected ResNet50 model along with early stopping criteria, which allowed investigation of several different configurations in a relatively short time. Training the model again with Schedule 8 resulted in the same accuracy although the one misclassified image was different. This can be attributed to the non-deterministic nature of training deep learning models. Running the model for longer with patience set to 50 epochs resulted in a slight deterioration in accuracy. Different experiments (in addition to schedule 10) with exponential decay for the learning rate did not yield better results and were abandoned. The CAM model surprisingly gave similar convergence and testing results with the one misclassified image corresponding to run 1 of the full model. This indicated that added Dense and Dropout layers did not have impact the outcome and could have been dropped in retrospect.

In addition to discriminating based on CSF hyperintensity, surprisingly a highly localized hot spot was also detected in the scalp (Figure 2C) due to fat suppression being employed with the FLAIR sequence. This was even though the activation map layer had 7×7 output which provides limited localization for input images of size 224×224. Better localization of the activation maps could possibly be achieved by taking the output of an earlier convolution layer with a larger activation map size. However, this would be further removed from our final model and perhaps not as indicative of the true areas of activation.

One prior work [1] used a modified VGG network for training and testing. Training time and epochs required were not reported. Although classification accuracy was high at 99.2%, spatial coverage of the brain was limited as first and last quarter of the images in a series were excluded in the sets used. Another work [2] used a network based on AlexNet operating on very low-resolution images of size 32×32. Training took 1000 epochs and 20 hrs to complete on a Nvidia Quadro K2200. Reported accuracy was 99.85%.

Although it's difficult to compare different works due to the differences in training and testing sets and classification tasks, the current work employed a relatively newer ResNet50 model which has been shown to perform better on medical imaging tasks [8]. Comparison with another high-performance CNN (InceptionV3) also established ResNet50 as the CNN of choice for this classification task. Testing accuracy may depend on the training, validation and testing sets employed, to the relative size of the testing set when compared with the training and validation set, and to the complexity of the images in terms of abnormalities. Images used for testing which deviate substantially from validation data could possibly be misclassified. Therefore, comparing accuracy across different works would be difficult for this task. The range of abnormalities encountered in brain images include craniotomies, lesions, hematomas, gliomas, cysts, encephalomalacia from ischemia, gliosis, in addition to artifacts resulting from motion, field inhomogeneity, implants like CSF shunts or neurostimulation devices like NeuroPace®. Therefore, to achieve close to 100% accuracy, a training set which cumulates images over time would be ideal for deployment in a real-world setting. Having said that, our trained model correctly classified images with pathology despite not having been trained on similar pathology (eg. Figure 2C).

The purpose of the current work was to establish a framework and methodology which allowed for fast training and deployment with very high accuracy. Faster learning is also conducive to repeated training of the network as new data get added to the training set. In addition, image resolution was maintained at 224×224 making it (at least theoretically) possible for the network to detect subtler changes when compared with prior works. Spatial coverage was high at 80% of the slices in a series. All this was achieved in an extremely short training time of ~9 min (on NVIDIA Tesla T4 GPU) due to employing an appropriate stopping criterion. Even when scaled to a larger data set, it's anticipated that training time will be relatively short. In addition, analysis of the network (which was not done in prior works) also provided some insight into the decision-making process of the network.

References

1. Ranjbar S, Singleton KW, Jackson PR, Rickertsen CR, Whitmire SA, Clark-Swanson KR, Mitchell JR, Swanson KR, Hu LS: A Deep Convolutional Neural Network for Annotation of Magnetic Resonance Imaging Sequence Type. J Digit Imaging 2020, 33(2):439-446.

2. Pizarro R, Assemlal HE, De Nigris D, Elliott C, Antel S, Arnold D, Shmuel A: Using Deep Learning Algorithms to Automatically Identify the Brain MR. Neuroinformatics 2019, 17(1):115-130.

3. Krizhevsky A, Sutskever I, Hinton GE: ImageNet Classification with Deep Convolutional Neural Networks. In: *Advances in Neural Information Processing Systems 25.* edn. Edited by Pereira F, Burges CJC, Bottou L, Weinberger KQ: Curran Associates, Inc.; 2012: 1097--1105.

4. Simonyan K, Zisserman A: Very Deep Convolutional Networks for Large-Scale Image Recognition. In.; 2014: arXiv:1409.1556.

5. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A: Going Deeper with Convolutions. In.; 2014: arXiv:1409.4842.

6. He K, Zhang X, Ren S, Sun J: Deep Residual Learning for Image Recognition. In.;2015:arXiv:1512.03385.

7. Chollet F: Xception: Deep Learning with Depthwise Separable Convolutions. In.; 2016: arXiv:1610.02357.

8. Ke, A., Ellsworth, W., Banerjee, O., Ng, A.Y., and Rajpurkar, P.: 'CheXtransfer: performance and parameter efficiency of ImageNet models for chest X-Ray interpretation': 'Proceedings of the Conference on Health, Inference, and Learning' (Association for Computing Machinery, 2021), pp. 116–124.

Appendix A: Sample training and testing images

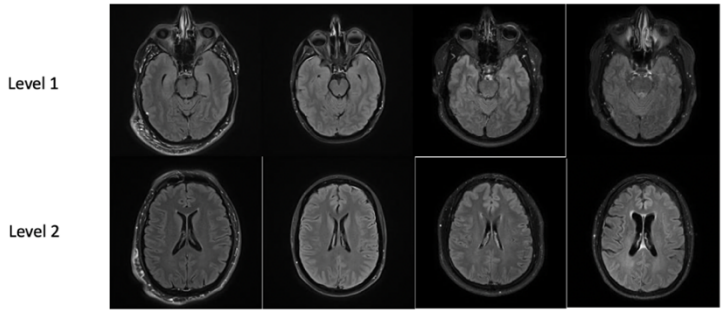


Figure 1: Sample post-contrast T2-weighted FLAIR images across 4 subjects and at two different levels in the brain from *training* set. Areas of hyperintensity around ventricles indicate pathology.

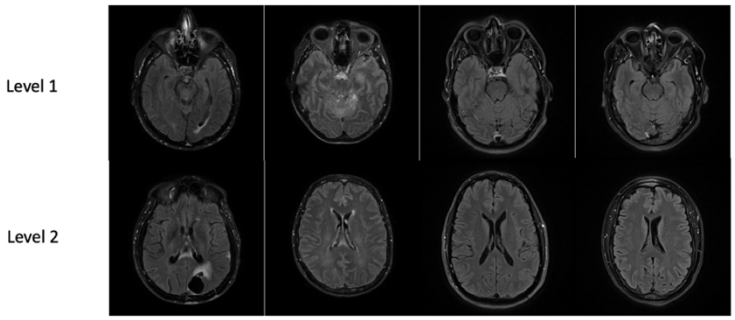
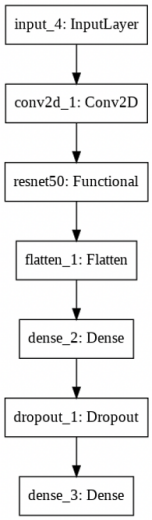


Figure 2: Sample post-contrast T2-weighted FLAIR images across 4 subjects and two different levels in the brain from the *testing* set. Areas of tissue hyperintensity and hypointensity outside ventricles and air spaces indicate pathology.

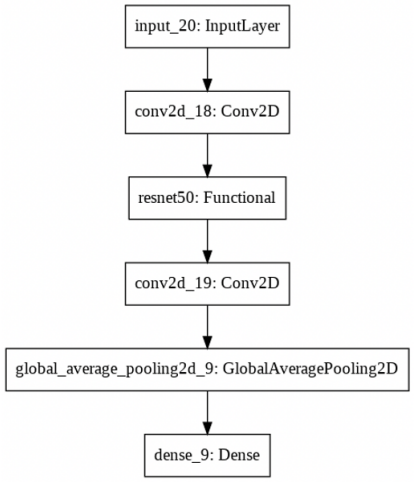
Appendix B: Training and Analysis Models



Layer (type)	Output Shape	Param #
input_4 (InputLayer)	[(None, 224, 224, 1)]	0
conv2d_1 (Conv2D)	(None, 224, 224, 3)	30
resnet50 (Functional)	(None, 7, 7, 2048)	23587712
flatten_1 (Flatten)	(None, 100352)	0
dense_2 (Dense)	(None, 1024)	102761472
dropout_1 (Dropout)	(None, 1024)	0
dense_3 (Dense)	(None, 4)	4100

Total params: 126,353,314
Trainable params: 112,755,746
Non-trainable params: 13,597,568

Figure 1: Schematic diagram of the trained model. L2 regularization was introduced for layers which allowed regularization in the base ResNet and Inception models.



Layer (type)	Output Shape	Param #
input_20 (InputLayer)	[(None, 224, 224, 1)]	0
conv2d_18 (Conv2D)	(None, 224, 224, 3)	30
resnet50 (Functional)	(None, 7, 7, 2048)	23587712
conv2d_19 (Conv2D)	(None, 7, 7, 2048)	4196352
global_average_pooling2d_9 (	(None, 2048)	0
dense_9 (Dense)	(None, 4)	8196

Total params: 27,792,290
Trainable params: 14,194,722
Non-trainable params: 13,597,568

Figure 2: Schematic diagram of the model used to obtain heat maps of class activation. A "dummy" convolution layer (con2d_19 in graph) of filter size 1×1, stride 1, padding 1) was introduced between the last ResNet50 convolution layer and global average pooling layer to circumvent an issue with TensorFlow 2 related to accessing output of the ResNet50 convolution layer.

Appendix C: Training and testing analysis for schedules described in Table 2.

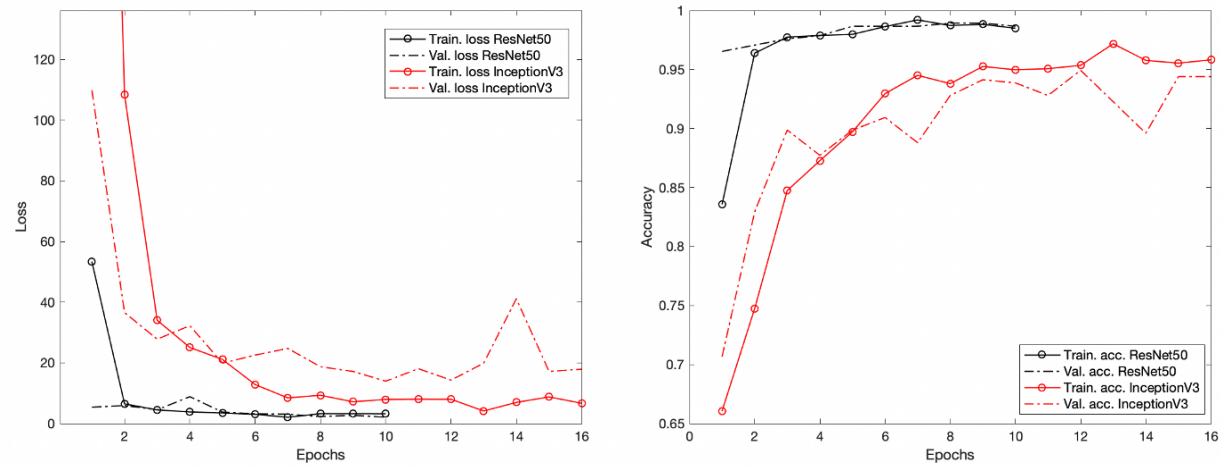


Figure 1: Comparing ResNet50 training and validation loss and accuracy (black) with InceptionV3 (red). See Table 2, schedules 1 and 2 for details. Clearly, ResNet50 outperformed InceptionV3 under these conditions for the medical imaging task.

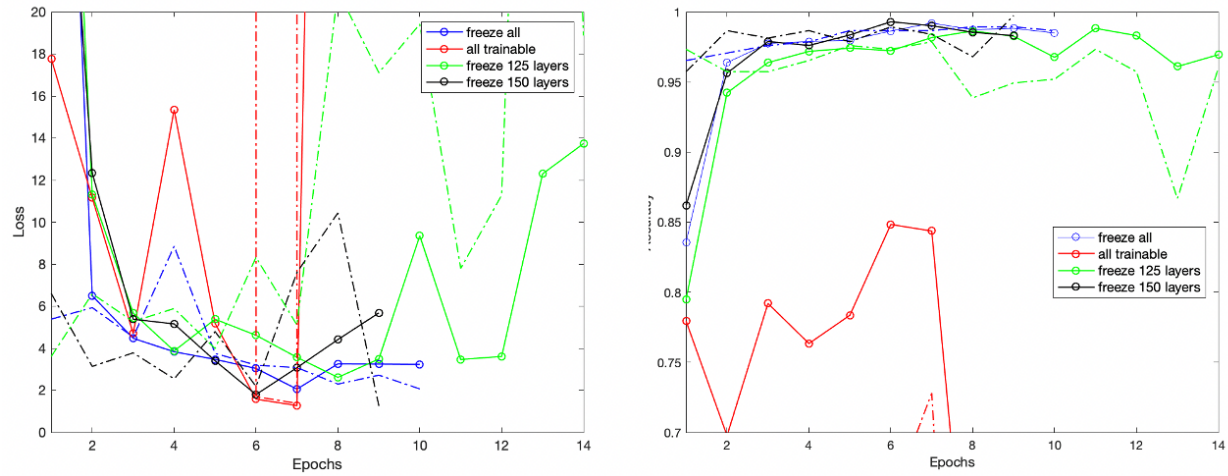


Figure 2: Training and validation loss (left) and accuracy (right) obtained while fine tuning of the base ResNet 50 model. Solid line indicates training loss or accuracy, while dashed line indicates validation loss or accuracy. Corresponds to schedules 1, 3, 4 and 5 in Table 2. Y-axis is zoomed in to show differences better. Based on the above, base model with all layers frozen and the first 150 layers frozen were considered as potential candidates for further analysis.

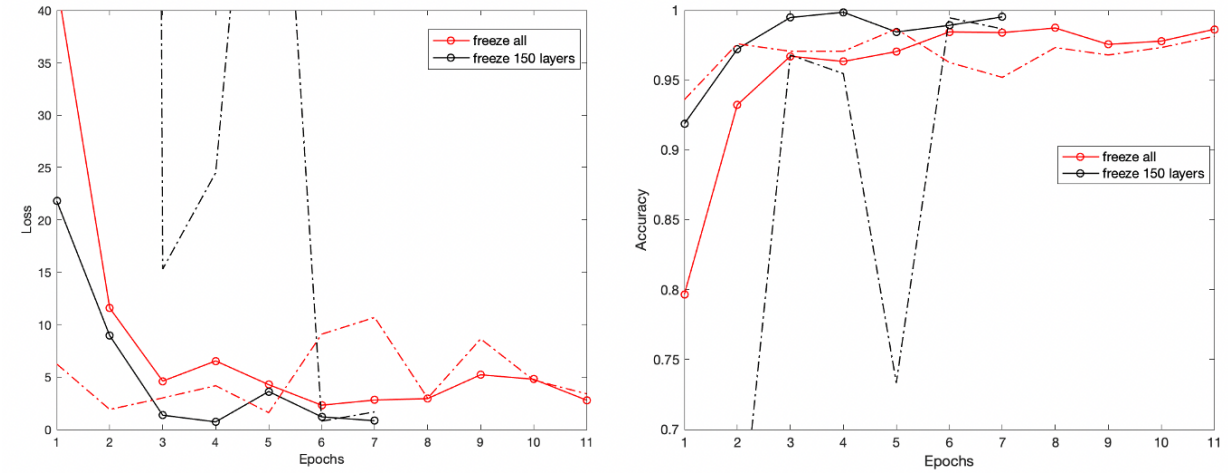


Figure 3: Training and validation loss and accuracy obtained while fine tuning of the base ResNet 50 model. Corresponds to schedules 6 and 7 in Table 2. Y-axis is zoomed in to show differences better. Although the validation loss and accuracy for the model with 150 layers frozen showed greater fluctuations initially, it converged faster and with higher accuracy for the established stopping criteria. Based on the above, the model with first 150 layers frozen was finally selected for further investigation.

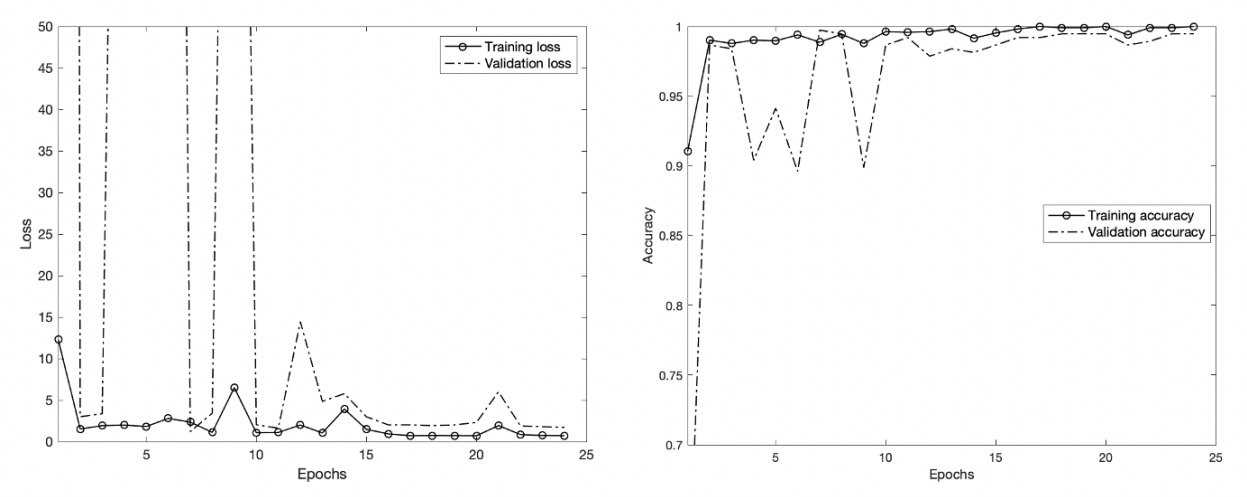


Figure 4: Training and validation loss and accuracy for the final candidate model with 150 frozen layers of the base ResNet50 model. Training schedule was #8 in Table 2.

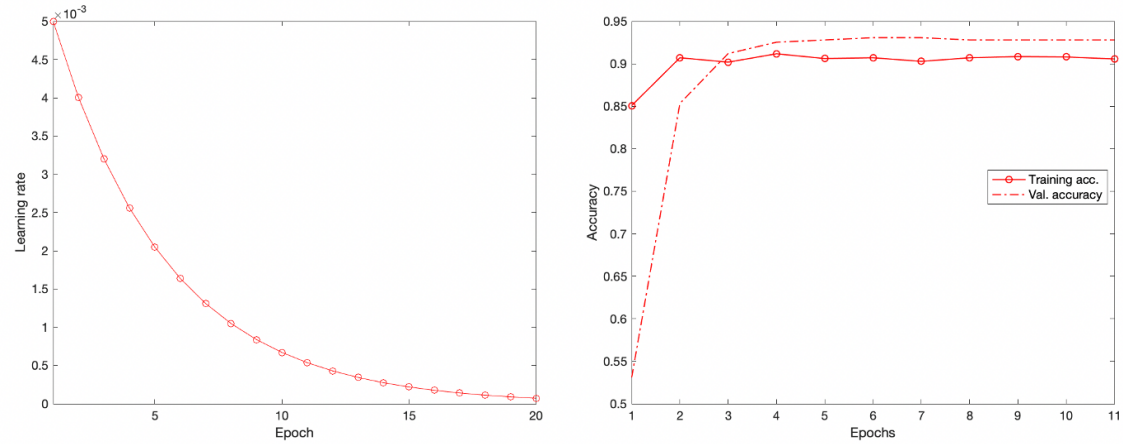


Figure 5: Once the model with highest accuracy was established, an adaptive learning rate (schedule 10 in Table 2) as shown on left was used to check for any further improvement on convergence and accuracy. However, while the training and validation accuracy curves were smoother, they exhibited sub-optimal asymptotic behavior leading to the stopping criteria kicking in relatively early and with lower training and validation accuracy than with constant lr = 0.001. See Discussion for more details.