Transfer Learning: Pre-trained VGG16 Architecture for Chest X-Ray Classification

Anirban Karmakar
M. Tech. CS1712

August 2020

Transfer Learning: Pre-trained VGG16 Architecture for Chest X-Ray Classification

DISSERTATION SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF

Master of Technology
in
Computer Science
by
Anirban Karmakar
[CS1712]

under the guidance of

Prof. Dipti Prasad Mukherjee
Electronics and Communication Sciences Unit

Indian Statistical Institute
Kolkata - 700108, India
August 2020

CERTIFICATE

This is to certify that the dissertation entitled **"Transfer Learning: Pre-trained VGG16 Architecture for Chest X-Ray Classification"** submitted by **Anirban Karmakar** to Indian Statistical Institute, in partial fulfillment for the award of the degree of **Master of Technology in Computer Science** is a bonafide record of work carried out by him under my supervision and guidance. The dissertation has fulfilled all the requirements as per the regulations of this institute and, in my opinion, has reached the standard needed for submission.

Dipti Prasad Mukherjee
Professor,
Electronics and Communication Sciences Unit,
Indian Statistical Institute,
Kolkata-700108, INDIA.

Acknowledgments

I would like to show my highest gratitude to my advisor, Prof. Dipti Prasad Mukherjee, Electronics and Communication Unit, Indian Statistical Institute, Kolkata, for his guidance and continuous support and encouragement. He has literally taught me how to do good research, and motivated me with great insights and innovative ideas.
My deepest thanks to all the teachers of Indian Statistical Institute, for their valuable suggestions and discussions which added an important dimension to my research work.
Finally, I am very much thankful to my parents and family for their everlasting supports.
Last but not the least, I would like to thank all of my friends for their help and support. I thank all those, whom I have missed out from the above list.

Anirban Karmakar
Indian Statistical Institute
Kolkata - 700108 , India.

Abstract

This thesis considers the task of thorax disease classification on Chest X-Ray images using transfer learning. The thorax or chest is a part of the anatomy of humans and various other animals located between the neck and the abdomen. The thorax contains organs including the heart, lungs, and thymus gland, as well as muscles and various other internal structures. Transfer learning from natural image datasets, particularly ImageNet, using models (VGG16, DenseNet, GoogLeNet etc.) and corresponding pretrained weights are used for deep learning applications to medical imaging. In this thesis, VGG16 network, which is pretrained on ImageNet data is explored. In Chest X-Ray14 dataset there are localized areas which are signs of abnormalities, whereas in ImageNet dataset, there is often a clear global subject of the image. Pretrained VGG16 had 1000 nodes in the output layer, one for each class. We change it to 14 nodes, one for each pathology: Atelectasis, Cardiomegaly, Effusion, Infiltration, Mass, Nodule, Pneumonia, Pneumothorax, Consolidation, Edema,Emphysema, Fibrosis, Pleural_Thickening, Hernia. We experiment with the strategy that CNN should act as a feature extractor. A performance evaluation shows that transfer offers little benefit to performance. We plot Receiver Operating Characteristic (ROC) curve for each of the pathologies. The area under the roc curve (AUROC) is calculated for each class. Average AUROC is calculated by taking the mean of all the classes. The average AUROC of our model is 0.715.

Keywords: VGG16, Transfer learning

Contents

Chapter 1

Introduction

Thorax diseases are very common in India. There are many reasons for that. Some of the reasons are outdoor air pollution, fire crackers, indoor air pollution due to the use of mosquito coils. The Chest X-Ray (CXR) is one of the most common radiological examinations in lung and heart disease diagnosis. Currently, interpreting CXRs mainly relies on professional knowledge and careful manual observation. Due to the complex pathologies and subtle texture changes of different lung lesion in images, radiologists may make mistakes even when they have long-term clinical training and professional guidance. Therefore, it is of importance to develop the CXR image classification methods to support clinical practitioners. The noticeable progress in deep learning has benefited many trials in medical image analysis, such as diseases classification [10],[16], image annotation [9] and so on. In this thesis, we investigate the CXR classification task using transfer learning. The advantage of using this approach is that we do not need to train the entire model. We take advantage of the pretrained weights of the model. These weights are reused. We use VGG16 as a feature extractor. All but the last feed-forward layer(s) of the network are frozen. The only weights that are trained are those in the last layers.

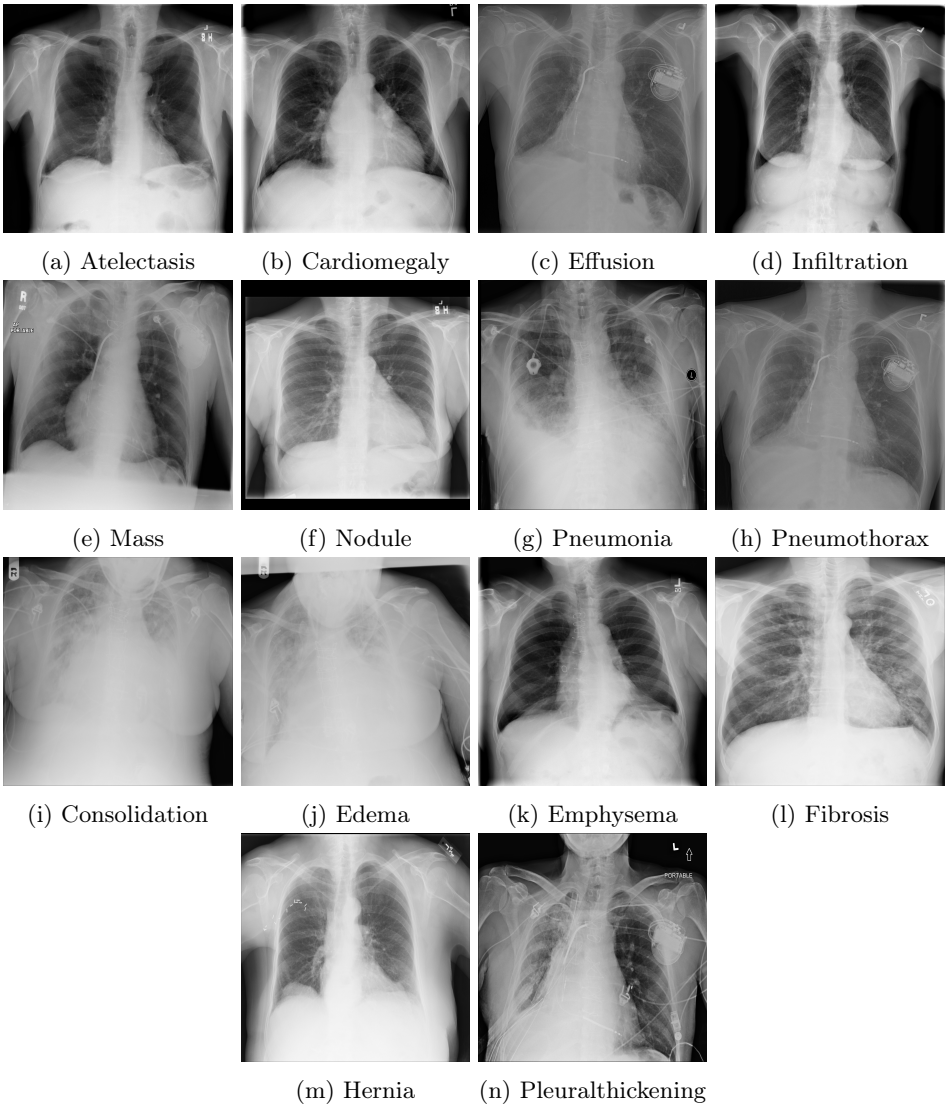
1.1 Objective

1. To provide radiologists and medical experts a low cost tool to cross check their interpretations.
2. Many people in our country do not have access to radiologist due to high cost. This tool can help them use telemedicine so that scarce medical resources can be accessed and used in a number of remote locations.

1.2 Related Work

1.2.1 Chest X-Ray datasets

The problem of Chest X-Ray image classification has been extensively explored in the field of medical image analysis. Several datasets have been released in this context. For example, the JSRT dataset [18] contains 247 Chest X-Ray images. The Shenzhen [1] Chest X-Ray set has a total of 662 images belonging to two categories (normal and tuberculosis (TB)). Among them, 326 are normal cases and 336 are cases with TB. The Indiana University Chest X-Ray collection [2] dataset has 3,955 radiology reports and the corresponding 7,470 Chest X-Ray images. Wang et al. [20] released the Chest X-Ray14 dataset, which is the largest Chest X-Ray dataset by far. Chest X-Ray14 collects 112,120 frontal-view Chest X-Ray images of 30,805 unique patients. Each radiography is labeled with one or more types of 14 common thorax diseases [3]. The 14 diseases are Atelectasis, Cardiomegaly, Effusion, Infiltration, Mass, Nodule, Pneumonia, Pneumothorax, Consolidation, Edema, Emphysema, Fibrosis, Pleural_Thickening, Hernia. Examples of 14 pathologies are shown in Figure 1.1.



(a) Atelectasis (b) Cardiomegaly (c) Effusion (d) Infiltration

(e) Mass (f) Nodule (g) Pneumonia (h) Pneumothorax

(i) Consolidation (j) Edema (k) Emphysema (l) Fibrosis

(m) Hernia (n) Pleuralthickening

Figure 1.1: Examples of pathologies [20]

1.2.2 Transfer Learning in medical imaging tasks

The use of ImageNet pretrained networks is becoming widespread in the medical imaging community. Lakhani et al. have [17] demonstrated the advantage of using ImageNet [11] pre-trained architectures for TB detection on small-scale datasets. They have used four datasets. This

includes two publicly available datasets maintained by the National Institutes of Health, which are from Montgomery County, Maryland [12], and Shenzhen, China [1]. The other two datasets are from Thomas Jefferson University Hospital, Philadelphia, and the Belarus Tuberculosis Portal maintained by the Belarus TB public health program [17]. These four datasets have 1007 Chest X-Rays in total. They have shown that two different deep convolutional neural networks, AlexNet [15] and GoogLeNet [19], pretrained on ImageNet dataset works better than AlexNet and GoogLeNet when they are not pretrained. Our work is different from the above work. We have studied VGG16 pretrained model on Chest X-Ray14 dataset [20] to diagnose 14 different thoracic pathologies: Atelectasis, Cardiomegaly, Effusion, Infiltration, Mass, Nodule, Pneumonia, Pneumothorax, Consolidation, Edema, Emphysema, Fibrosis, Pleural_Thickening, Hernia.

1.3 Overview of the Proposed Approach

In computer vision applications, deep learning models are rarely trained from scratch, but instead transfer learning is used. In order to use a CNN pre-trained from ImageNet, the last fully-connected layer of the pretrained model is modified. Figure 1.2 is a diagram of VGG16 architecture which we have used in our thesis. The last layer (the last Dense layer as shown in Figure 1.2) is modified to the number of classes of the given problem. The stack of convolution layers (which are shown in Figure 1.2 as Conv1-1, Conv1-2 etc) are kept frozen meaning we do not train them as they are already trained by ImageNet data and we only reuse the weights of these layers. We have only trained the fully connected classifier (the last three Dense layers as shown in Figure 1.2) by NIH Chest X-Ray14 [20] dataset.

Figure 1.2: Diagram of VGG16 architecture [4]

1.4 Contribution

We have used transfer learning for Chest X-Ray classification. We have done the following things:

- VGG16 had 1000 way output to classify 1000 classes. We have changed it to 14 way output to classify 14 classes as we have 14 pathologies.
- Initially VGG16 has a stack of convolutional layers and it has a fully connected classifier on top of that. We have not trained the convolutional layers and only trained the fully connected classifier.
- We have studied the performance of the proposed modification of VGG16 using Chest X-Ray14 [20] dataset.

1.5 Organization of the thesis

This thesis is divided into 4 chapters. The layout of every chapter is given in the following:

- Chapter 1: Contains an introduction, objective and an overview of the proposed approach to solve the problem.
- Chapter 2: Presents the overall approach, motivation, architecture used, experimental setup, training, validation and test.

- Chapter 3: Presents the training, validation and test result and discussion on results.
- Chapter 4: This chapter concludes our work. This chapter contains the summary of our work done and future directions.

Chapter 2

Methodology

2.1 Overall Approach

We adopt an approach known as transfer learning. We have not trained an entire Convolutional Neural Network from scratch. All but the last feed-forward layers of the network were frozen. The only weights that are trained are those in the last layers [14].

2.2 Motivation

Motivation of using transfer learning comes from the following fact that the earlier layers of a Convolutionl Neural Network contain more generic features, but later layers of the Convolutionl Neural Network becomes progressively more specific to the details of the classes contained in the original dataset.

2.3 Description of Architecture

VGG16 architecture have been used as a backbone. The VGG16 architecture was introduced by Simonyan and Zisserman in their 2014 paper [13]. We have kept the VGG16 architecture same except the last layer. We have changed the 1000 way output to 14 way output as we have 14

pathologies to classify. Description of layers and the output dimension for each layer is shown in Table 2.1. The modified VGG16 architecture is shown in Figure 2.1.

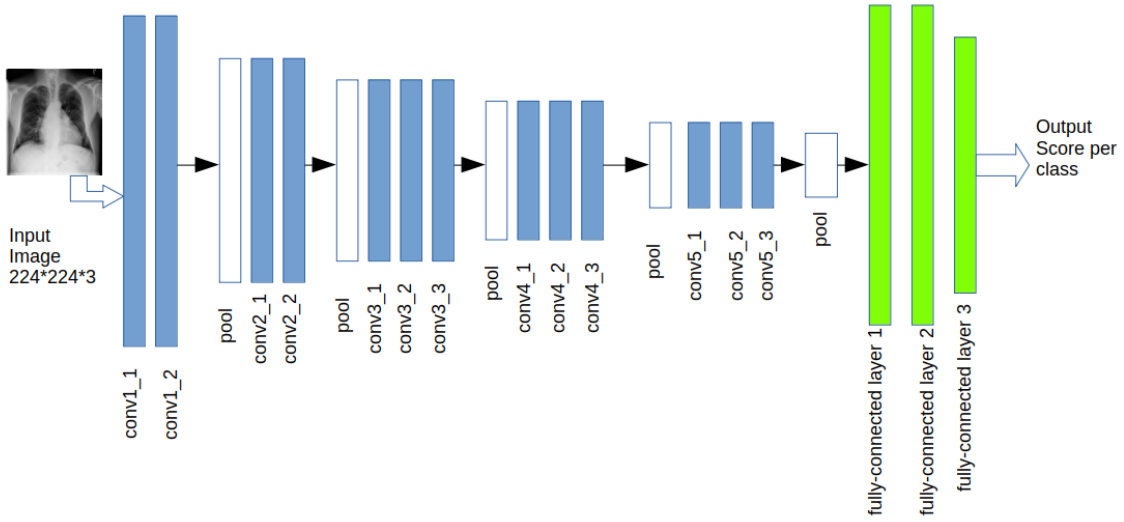


Figure 2.1: Modified VGG16 architecture

All the convolutional layers have kernel size 3×3 with pad = 1 and stride = 1. All pool layers shown in Figure 2.1 are maxpool layers with kernel size 2×2 with stride = 2. All convolutional layers and fully-connected layers have ReLU activation function. The input to the network is an image of dimension 224×224×3 meaning height 224, width 224 and three color channels R, G, B.

Layers	Output dimension (height × width × no of filters)
conv1_1, conv1_2	224 × 224 × 64
pool	112 × 112 × 64
conv2_1, conv2_2	112 × 112 × 128
pool	56 × 56 × 128
conv3_1, conv3_2, conv3_3	56 × 56 × 256
pool	28 × 28 × 256
conv4_1, conv4_2, conv4_3	28 × 28 × 512
pool	14 × 14 × 512
conv5_1, conv5_2, conv5_3	14 × 14 × 512
pool	7 × 7 × 512
fully-connected layer 1	1 × 1 × 4096
fully-connected layer 2	1 × 1 × 4096
fully-connected layer 3	1 × 1 × 14

Table 2.1: Description of layers and output dimension for each layer

2.4 Datasets

In transfer learning there are two tasks: the "source" task, generally a large dataset on which pre-training is performed (e.g., ImageNet, which contains 1.2 million images with 1000 categories), and the "target" task of interest. In this work, source refers to the dataset or task with which the network is first trained, and target refers to the dataset or task with which the network is fine-tuned. The following describes the datasets used in this study:

2.4.1 Source Dataset

- ImageNet dataset: ImageNet [11] is a large dataset of annotated photographs intended for computer vision research. There are a little more than 14 million images in the dataset, a little more than 21 thousand groups or classes and a little more than 1 million images that have bounding box annotations. Figure 2.2 is an example of the ImageNet data.

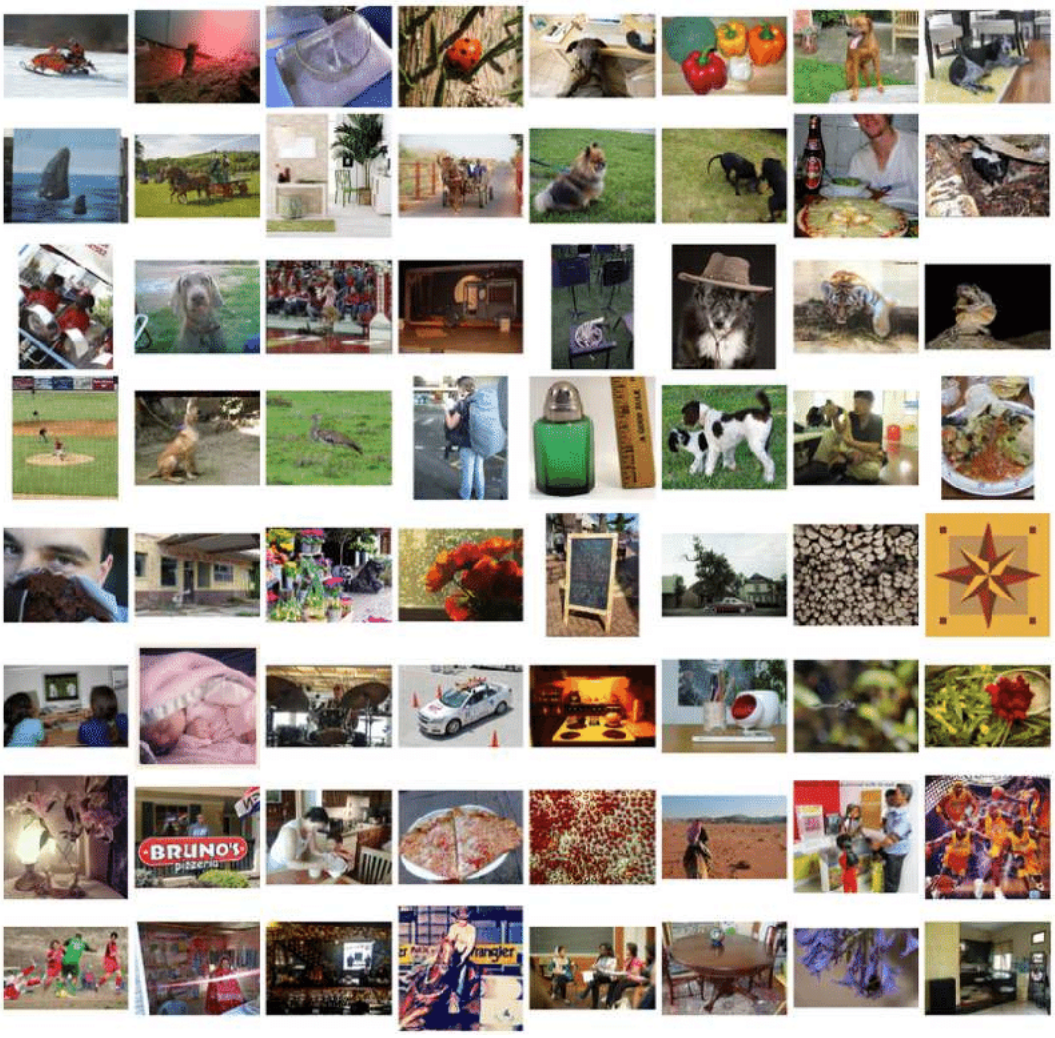


Figure 2.2: Examples of the ImageNet dataset [5]

2.4.2 Target Dataset

- Chest X-Ray14 dataset: Chest X-Ray14 [3] collects 112,120 frontal-view images of 30,805 unique patients. 51,708 images of them are labeled with up to 14 pathologies, while the others are labeled as "No Finding".

2.5 Experimental Setup

2.5.1 Multilabel Setup:

Each image is labeled with a 14-dim vector L = [l_1,l_2,l_3,...,l_C] in which $l_C \in \{0, 1\}$, $C = 14$. l_C represents whether there is any pathology, i.e., 1 for presence and 0 for absence.

2.6 Data Resize and Mapping

We use the following data resizing and mapping for training as well as testing.

1. Resize the images from 1024×1024 to 224×224.
2. Map each pixel value from 0 to 255 to 0 to 1

2.7 Training and Testing/Validation

2.7.1 Data Split

In our experiment, we randomly shuffled the dataset into three subsets: 70% for training, 10% for validation and 20% for testing.

2.7.2 Loss Function

In this thesis we have used binary cross entropy (BCE) loss because in binary cross entropy loss the loss is small for correct classification

and large for misclassification. This is a medical imaging task and we want to penalise our proposed model for misclassification. I is the input image. $\tilde{p}_g(c|I)$ is the probability score of I belonging to the cth class, $c \in \{1, 2, ..., C\}$. We optimize our model by minimizing the binary cross-entropy (BCE) loss:

$$Loss = \frac{-1}{C} \sum_{c=1}^{C} l_c log(\tilde{p}_g(c|I)) + (1 - l_c) log(1 - \tilde{p}_g(c|I)) \tag{2.1}$$

where l_c is the groundtruth label of the cth class, C is the number of pathologies.

2.7.3 Optimizer

We have 78468 training images, so if we use a typical Gradient Descent optimization technique, we have to go over all the training images before updating the parameters, and it has to be done for every iteration until the minima is reached. Hence, it becomes computationally very expensive to perform. That is why we have used Stochastic Gradient Descent so that we can have small batches and we go over the those small batches before updating the parameters. We have used Stochastic Gradient Descent (SGD) with momentum. Momentum helps to accelerate gradients in the right direction.

2.8 Experiment

For training, we have resized the original images to 224×224. We have mapped each pixel values from 0 to 255 to 0 to 1. We have optimized our network using SGD with batch size of 30. We have trained the classifier for 50 epochs. The learning rate is 0.00001 and the momentum is 0.9. During validation and testing, images are resized to 224×224.

Batch size 30 is used during validation. Batch size 2 is used during test. Implementation is done using the PyTorch framework [6].

2.9 Procedure

1. Pre-trained VGG16 model have been downloaded from [7].
2. Weights of the convolutional layers have not been trained.
3. The number of outputs of the classifier have been set equal to the number of classes.
4. Only the classifier have been trained.

2.10 Evaluation Metrics

The quality our proposed model is evaluated in terms of two measures: accuracy, area under receiver operating characteristics (AUROC) curve. Receiver Operating Curve (ROC) is drawn using scikit-learn [8]. The accuracy is the ratio of number of correctly classified samples to total samples. ROC curve is the graphical plot of true positive rate (TPR) vs false positive rate (FPR) of a binary classifier. Say, in a binary classifier the outcomes are labeled either as positive (p) or negative (n). There are four possible outcomes from a binary classifier. If the outcome from a prediction is p and the actual value is also p, then it is called a true positive (TP); however if the actual value is n then it is said to be a false positive (FP). Conversely, a true negative (TN) has occurred when both the prediction outcome and the actual value are n, and false negative (FN) is when the prediction outcome is n while the actual value is p. True positive rate (TPR), measures the proportion of positives (p) that are correctly identified

$$TPR = \frac{TP}{TP + FN} \tag{2.2}$$

False positive rate (FPR), measures the proportion of negatives (n) incorrectly identified as positives (p)

$$FPR = \frac{FP}{FP + TN} \tag{2.3}$$

Figure 2.3: Example of correctly classified sample

Class	Probability Score
Atelectasis	**0.6784**
Cardiomegaly	0.0372
Effusion	0.0367
Infiltration	0.1369
Mass	0.0170
Nodule	0.0283
Pneumonia	0.0074
Pneumothorax	0.0109
Consolidation	0.0168
Edema	0.0041
Emphysema	0.0069
Fibrosis	0.0058
Pleural_Thickening	0.0113
Hernia	0.0024

Table 2.2: Probability score for each class for the example shown in Figure 2.3

The true label of Figure 2.3 as given in the Chest X-Ray14 [20] is Atelectasis. The image is fed to our proposed model. We have got a 14-dim

ouput vector z = [z_1,z_2,z_3,. . . ,z_{14}], where $\forall i \in \{1, 2, 3, ..., 14\}$, $z_i \in \mathbb{R}$. The softmax function $\sigma : \mathbb{R}^{14} \to \mathbb{R}^{14}$ is defined by the formula

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{14} e^{z_j}} \tag{2.4}$$

for i = 1,2,...,14. We have calculated the probabilities for each class using equation 2.4. We have showed probability score for each class in Table 2.2. Probability score is the highest for Atelectasis. So, we have considered Atelectasis class as the output of our proposed model. So in this case the true label of our image is the same as the output. This is an example of a correctly classified sample or True Postive (TP).

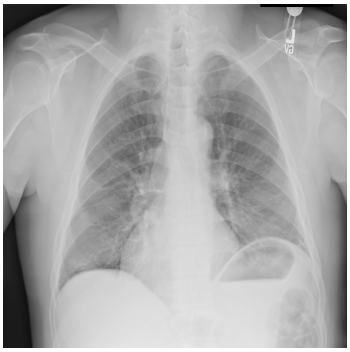


Figure 2.4: Example of incorrectly classified sample

The true label of Figure 2.4 as given in the Chest X-Ray14 [20] is Pneumonia. The image is fed to our proposed model. We have got a 14-dim ouput vector z = [z_1,z_2,z_3,. . . ,z_{14}], where $\forall i \in \{1, 2, 3, ..., 14\}$, $z_i \in \mathbb{R}$. We have calculated the probability score for each class using equation 2.4. We have showed probability score for each class in Table 2.3. Output probability is the highest for Atelectasis. So, we have considered Atelectasis class as the output of our proposed model. So in this case the true label of our image is not the same as the model output. This is an example of an incorrectly classified sample.

Class	Probability Score
Atelectasis	**0.7382**
Cardiomegaly	0.0273
Effusion	0.0170
Infiltration	0.0815
Mass	0.0246
Nodule	0.0545
Pneumonia	0.0084
Pneumothorax	0.0069
Consolidation	0.0125
Edema	0.0056
Emphysema	0.0046
Fibrosis	0.0065
Pleural_Thickening	0.0109
Hernia	0.0014

Table 2.3: Probability score for each class for the example shown in Figure 2.4

2.11 Summary

In this section the transfer learning approach has been discussed. How the traditional VGG16 architecture have been modified to solve our problem. How the data is resized and mapped, how the data is divided into three subsets such as training, testing and validation have been discussed. There are other essential things such as the loss functions, optimizer, how the experiment is done and also the details about the evaluation measures.

Chapter 3

Results and Discussion

3.1 Training, Validation and Test data

1. No of Training images: 78468
2. No of Validation images:11219
3. No of Test images: 22433

3.2 Initialization Techniques

For the convolution layers the pretrained weights have been used. For the classifier we have learned the weights. We have used default PyTorch initialization.

3.3 Parameters

We have not trained the entire VGG16 network. Only the fully connected classifier part is trained. The fully connected classifier part is shown in Figure 3.1. The number of parameters per layer which we have trained are shown in Table 3.1.

Layer	No of parameters
Fully-connected Layer 1	102,764,544
Fully-connected Layer 2	16,781,312
Fully-connected Layer 3	61,455

Table 3.1: No of trainable parameters of the last three fully-connected layers as shown in Figure 2.1

The number of parameters trained = 119607311

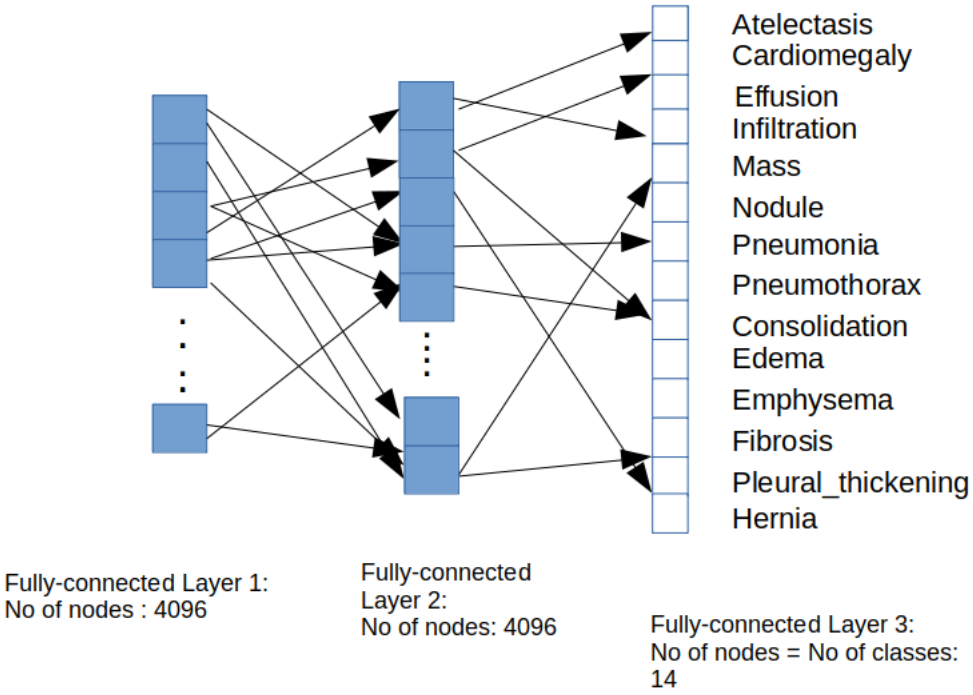


Figure 3.1: Fully Connected Classifier of our proposed model (last three fully-connected layers of the architecture shown in Figure 2.1)

3.4 Training, Validation and Test Result

3.4.1 Experimental Details

Batch Size = 30
No of images for training = 78468
No of images for validation = 11219
Optimizer Used = SGD
Momentum Used = 0.9
Learning Rate = 0.00001
Number of epochs = 50

3.4.2 Training, Validation Loss and Accuracy

$$Loss\ per\ batch = Average\ loss\ per\ batch \times batch\_size \tag{3.1}$$

Total loss is calculated by finding out the *Loss per batch* by equation 3.1 for every batch and summing them up.

$$Loss\ per\ epoch = \frac{Total\ loss}{Number\ of\ images\ per\ epoch} \tag{3.2}$$

$$Accuracy\ per\ epoch = \frac{No\ of\ correctly\ classified\ images\ per\ epoch}{Total\ no\ of\ images\ per\ epoch} \times 100\% \tag{3.3}$$

Loss per epoch and *Accuracy per epoch* are computed by equations 3.2 and 3.3 respectively. Final loss and accuracy is the loss and accuracy after the last epoch. Final training and validation loss is shown in Table 3.2. Final training and validation loss is shown in Table 3.3. Training and validation accuracy per epoch is shown in Figure 3.2. There are fluctuations in the accuracy curve. The reason could be that the size of our training data is small. So the training accuracy curve is fluctuating.

Some of the images in the validation set may be classified randomly by our proposed model and this random classification causes fluctuations in the validation accuracy curve. Training and validation loss per epoch is shown in Figure 3.3. Training and validation loss both are decreasing which means that our proposed model is learning from the training images and it is able to classify the unseen images of the validation set.

Training loss	1.2183
Validation loss	1.24310

Table 3.2: Final Training and Validation Loss

Training Accuracy	64.442 %
Validation Accuracy	64.114 %

Table 3.3: Training and Validation Accuracy

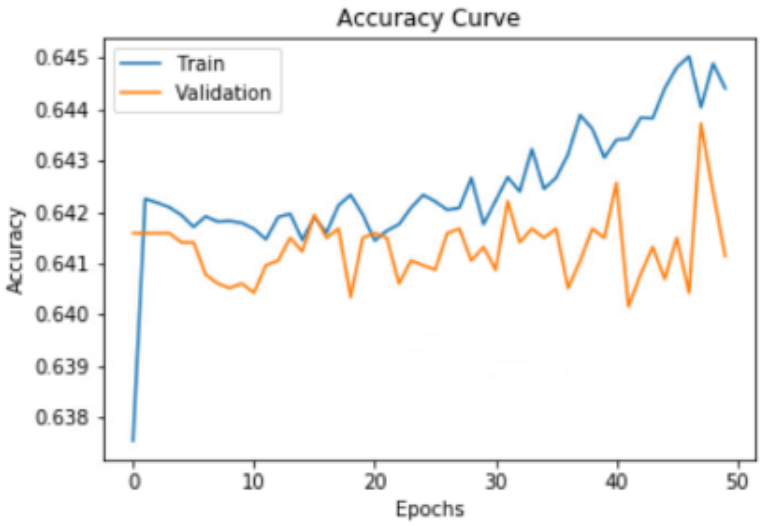


Figure 3.2: Accuracy vs epoch

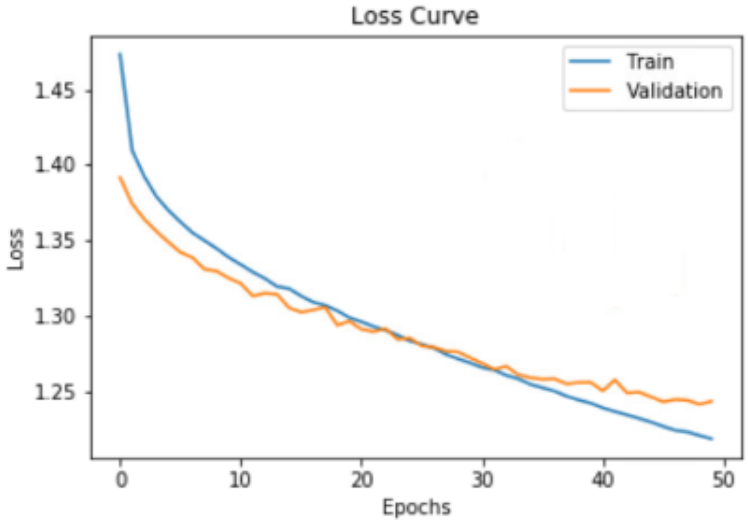


Figure 3.3: Loss vs epoch

3.4.3 Test Results

$$Test\ Loss\ per\ batch = Average\ loss\ per\ batch \times batch\_size \quad (3.4)$$

Total loss is calculated by finding out the *Test Loss per batch* by equation 3.4 for every batch and summing them up.

$$Test\ Loss = \frac{Total\ loss}{Number\ of\ images} \quad (3.5)$$

$$Test\ Accuracy = \frac{No\ of\ correctly\ classified\ images}{Total\ no\ of\ images} \times 100\% \quad (3.6)$$

Test Loss and Test Accuracy are computed using the equations 3.5 and 3.6 respectively and shown in Table 3.4.

Test Accuracy	63.843 %
Test Loss	1.27245

Table 3.4: Test Loss and Accuracy

3.5 Numerical Accuracy (in terms of Area Under the ROC curve)

3.5.1 AUROC plot

We have 14 pathologies. ROC curve is a graphical plot where x-axis is False Positive Rate (FPR) and y-axis is True Positive Rate (TPR). Consider one class Edema. Now, we change the threshold value from 0 to 1 and calculate number of True Positive (TP), False Positive (FP), True Negative (TN), False Negative (FN) for this one class. If the outcome of our proposed model is Edema and the actual class is Edema then we have True Positive (TP); if the outcome of our proposed model is Edema but the true class is some class other than Edema then it is said to be a False Positive (FP). When the outcome is some class other than Edema and the actual class is also some class other than Edema then we have True Negative (TN). When the actual class is Edema and the output of our proposed model is some class other than Edema then we have False Negative (FN). After we have calculated these four values we calculate True Positive Rate (TPR) and False Positive Rate (FPR) for this class using equations 2.2 and 2.3 and plot them for each threshold value. We have similarly drawn ROC curves for other pathologies. We have plotted ROC curve for each pathology and it is shown in Figure 3.4. AUROC is the area under the ROC curve. AUROC is shown in Table 3.6 for each class. Average AUROC is the mean of AUROC for all the classes. The average AUROC is 0.715.

Class	No of images per class
Atelectasis	2420
Cardiomegaly	502
Effusion	1832
Infiltration	2700
Mass	657
Nodule	731
Pneumonia	86
Pneumothorax	539
Consolidation	274
Edema	116
Emphysema	206
Fibrosis	174
Pleural_Thickening	247
Hernia	21

Table 3.5: No of images per class in test set

Class	AUROC
Atelectasis	0.46
Cardiomegaly	0.82
Effusion	0.84
Infiltration	0.50
Mass	0.71
Nodule	0.58
Pneumonia	0.69
Pneumothorax	0.77
Consolidation	0.76
Edema	0.83
Emphysema	0.83
Fibrosis	0.68
Pleural_Thickening	0.71
Hernia	0.82

Table 3.6: AUROC for each pathology

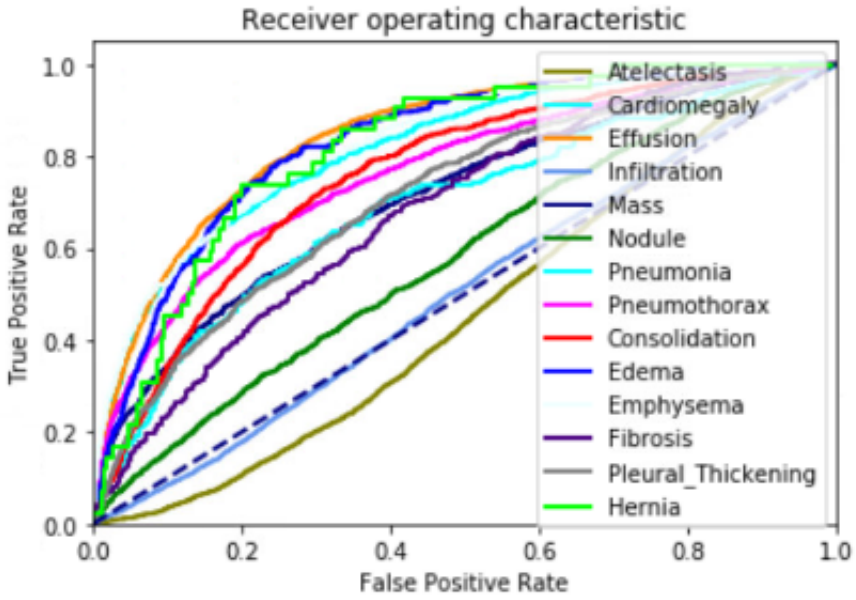


Figure 3.4: ROC

3.5.2 Discussion on results

The accuracy of our proposed model is very poor. The reason for this is we have used VGG16 architecture which is pretrained on the ImageNet data. We are reusing ImageNet features. But ImageNet features and Chest X-Ray image features are quite different. In Chest X-Ray images the lesion areas can be very small, and the position is unpredictable. Sometimes there are local white opaque patches which are signs of abnormalities. This is in contrast with ImageNet dataset, where there is often a clear subject for the image.

Chapter 4

Conclusions

4.1 Conclusions

In this thesis, we have used transfer learning approach for Chest X-Ray classification. We have investigated the performance of a VGG16 architecture which is pretrained by the ImageNet data. We have modified the last layer from 1000 way output to 14 way output. We have reused the weights for the convolution layers and trained the fully connected classifier by Chest X-Ray14 [20] data.

4.2 Future Directions

As it can be seen from the results that the accuracy is very poor and the model cannot at all be deployed in the real world. That is where lies the motivation to do future research on this topic. With the same dataset and the same transfer learning approach we plan to evaluate every state of the art architecture so that higher accuracy can be achieved.

Bibliography

[1] Last accessed as on August 2020. URL: https://lhncbc.nlm.nih.gov/publication/pub9931.

[2] Last accessed as on August 2020. URL: https://www.kaggle.com/raddar/chest-xrays-indiana-university.

[3] Last accessed as on August 2020. URL: https://www.kaggle.com/nih-chest-xrays/data.

[4] Last accessed as on August 2020. URL: https://neurohive.io/en/popular-networks/vgg16/.

[5] Last accessed as on August 2020. URL: https://www.researchgate.net/figure/Examples-in-the-ImageNet-dataset_fig7_314646236.

[6] Last accessed as on August 2020. URL: https://pytorch.org/tutorials/beginner/transfer_learning_tutorial.html.

[7] Last accessed as on August 2020. URL: https://download.pytorch.org/models/vgg16-397923af.pth.

[8] Last accessed as on August 2020. URL: https://scikit-learn.org/stable/auto_examples/model_selection/plot_roc.html.

[9] S. Albarqouni et al. *Aggnet: deep learning from crowds for mitosis detection in breast cancer histology images*. IEEE transactions on medical imaging, vol. 35, no. 5, pp. 1313–1321. 2016.

[10] M. Anthimopoulos et al. *Lung pattern classification for interstitial lung dis- eases using a deep convolutional neural network*. IEEE transactions on medical imaging, vol. 35, no. 5, pp. 1207–1216. 2016.

[11] Deng and Jias. *ImageNet: A large-scale hierarchical image database*. CVPR, IEEE Conference. 2009.

[12] S. Jaeger et al. *Two public chest x-ray datasets for computer-aided screening of pulmonary diseases*.

[13] Andrew Zisserman Karen Simonyan. "Very Deep Convolutional Networks for Large-Scale Image Recognition". In: (2014). DOI: https://arxiv.org/abs/1409.1556.

[14] Andrej Karpathy. *Cs231n convolutional neural networks for visual recognition*. URL: http://cs231n.github.io/transfer-learning/.

[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton. *Imagenet classification with deep convolutional neural networks*. Advances in neural information processing systems, 2012, pp. 1097–1105. 2012.

[16] P. Kumar, M. Grewal, and M. M. Srivastava. *Boosted cascaded convnets for multilabel classification of thoracic diseases in chest radiographs.* arXiv preprint arXiv:1711.08760. 2017.

[17] P. Lakhani and S. Baskaran. *Deep learning at chest radiography: automated classification of pulmonary tuberculosis by using convolutional neural networks.* Radiology 284.2 574-582. 2017.

[18] J. Shiraishi et al. *Development of a digital image database for chest radiographs with and without a lung nodule: receiver operating characteristic analysis of radiologists detection of pulmonary nodules.* American Journal of Roentgenology, vol. 174, no. 1, pp. 71–74. 2000.

[19] C. Szegedy et al. *Going deeper with convolutions.* in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 1–9. 2015.

[20] Xiaosong Wang et al. *ChestX-ray8: Hospital-scale Chest X-ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseases.*