Identification of Pneumothorax in X-ray images using Convolutional neural network

MSc Research Project
Programme Name: MSc DAD-A 2019-20

Aditya Sharma
Student ID: 18198821

School of Computing
National College of Ireland

Supervisor: Christian Horn

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Aditya Sharma…………………………………………………

Student ID: 18198821……………………………………………………………..……

Programme: MSC-DAD-A……………………………… **Year:** 2019-20………..

Module: Research Project…………………………………………………….………

Supervisor: Christian Horn………………………………………………………………….………

Submission Due Date: 17-08-2020……………………………………………………….………

Project Title: Identification of Pneumothorax in X-ray images using Convolutional neural network.

Word Count: 6738 **Page Count** 21

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: aditya sharma…………………………………………………………………………

Date: 16-08-2020…………………………………………………………………………

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Identification of Pneumothorax in X-ray images using Convolutional neural network

Aditya Sharma
Student ID - 18198821

Abstract

Pneumothorax is a condition in which air gets accumulated in the pleural space of thorax, which lies between chest and lungs. It causes the air which is present in the lungs during breathing to come out of lung and gets trapped between chest and lungs, in the pleural space of chest which causes chest pains, breathing difficulty and can also lead to total collapse of one lung in case of tension pneumothorax. Therefore, this should be diagnosed early so that the treatment could get started soon. Neural networks have proved to be effective in image classification tasks and many researchers have proved that chest anomalies like pneumonia, pneumothorax etc, which are diagnosed by radiologists, can be detected by training a neural network model. In this research, ResNet-50 neural network will be trained on a dataset of chest x-ray images having more than 10,000 images to create a model which would predict the presence of pneumothorax from chest x-ray images.

1 Introduction

Pneumothorax is an uncommon condition in which air gets collected between the lung and chest wall, in the pleural space. Symptoms of pneumothorax includes chest pain and shortness of breath. In some cases, a one-way valve could be formed due to which the air in pleural space increases which causes shortage of oxygen and low blood pressure, this is fatal in severe cases. The air leaks from lungs and gets accumulated in the pleural space. This occurs due to various reasons such as chest injury in which a blunt or penetrating injury to chest causes the air to leak out from the lungs.

Pneumothorax can be divided into two categories, traumatic pneumothorax and non-traumatic pneumothorax. Non-traumatic pneumothorax do not occur from injuries but other conditions such as underlying lung diseases such as pneumonia, cystic fibrosis (which is a hereditary diseases in which mucus is produced by the body which clogs the lungs), chronic obstructive pulmonary disease (COPD) which blocks the airflow in lungs; ruptured air blisters, in which small air blisters known as blebs which are present on top of lungs ruptures which causes air to leak from them and get accumulated in pleural space. People who are very long and thin also gets pneumothorax, but not very severe. Non-traumatic pneumothorax is not fatal in most of the cases and causes mild discomfort like difficulty in breathing and chest pains. This type of pneumothorax is caused in people having underlying lung problems.

Traumatic pneumothorax is the result of chest injuries in which the chest wall is pierced or damaged from accident and as a result air gets accumulated from leakage from lungs. The ribs getting injured from accident may also cause traumatic pneumothorax as they pierce the lungs in some cases which results in air getting leaked from lungs.

Tension pneumothorax is caused when there is significant damage to lungs and there is a difficulty in breathing. Any type of pneumothorax, traumatic or non-traumatic can result in tension pneumothorax. Tension pneumothorax is a medical emergency and requires immediate assistance. It causes total collapse of lung and is fatal. The trachea also gets shifted towards the normal lung and diaphragm gets pushed downwards too, which causes the increase in the pleural pressure and distention of jugular vein.

Pneumothorax can be identified from chest x-rays. The air that gets accumulated in the pleural space is dark in the x-ray images. Various aspects of x-ray images in determining pneumothorax are used by doctors and radiologists such as change of colour intensity where pneumothorax is present, difference in the right and left lungs in x-ray and a dark line present in the lung having pneumothorax at top corner as air reaches the boundaries of pleural space. The air gets accumulated in the lateral space of thorax as air tries to reach the highest area.

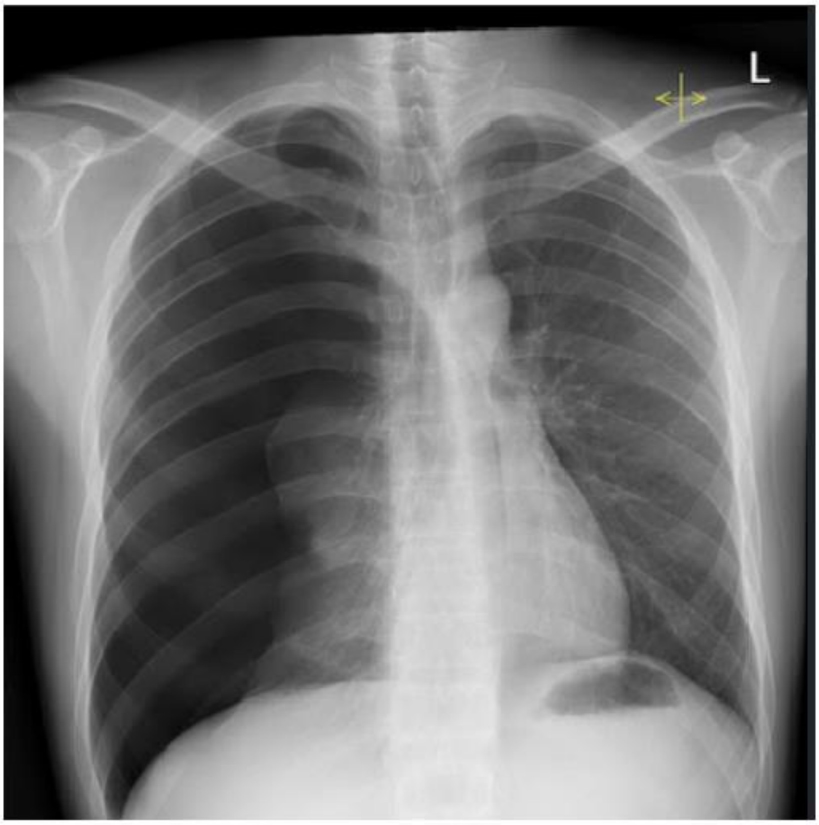


Fig: 1 - X-ray having pneumothorax [1]

[1] Fig 1: Source: https://www.kaggle.com/vbookshelf/pneumothorax-chest-xray-images-and-masks

The above image is having pneumothorax, in the left lung there is a darker black area which implies the presence of air. The black in the left lung which is separating the light black and dark black area is also a positive sign of pneumothorax. In the case of tension pneumothorax, one lung which is having the pneumothorax would move slightly towards and push the other lung. It leads to total collapse of one lung.

Convolutional neural networks (CNN) are used specifically for image classification tasks, due to their huge architecture, various properties of the images are captured by CNNs which are helpful in classifying the images. ResNet-50 is a special type of CNN which has a residual block implemented in the network to avoid gradient vanishing problem which stops the learning process of CNN. ResNet-50 performed very well on the ImageNet dataset and was the winner of ILSVRC 2015 image classification challenge.

The paper is divided in eight sections in which Intr0duction was the first section followed by dataset description in which the details of the data which would be used in the research is discussed. After the related work of many researchers is discussed. Next section is research methodology in which working of CNN is discussed followed by the discussion on design of ResNet-50. After that section implementation is discussed which is followed by results, conclusion and future work.

2 Related Work

Karim et al. (2020) have tested various CNN models VGG-16, VGG-19, ResNet-34, ResNet-18, DenseNet-161 and DenseNet-201; to detect covid-19. Three datasets are used in the research - The COVIDx v1.0 dataset had a total of 5,941 CXR images from 2,839 patients based on the COVID-19 image dataset curated by Joseph P., et al.[2]; COVIDx v2.0 dataset retrieved from RSNA[3] pneumonia detection challenge; third dataset is collected from Italian Radiological Case[4](provided by Dr. Fabio Macori). Many data augmentations are applied in data pre-processing like contrast enhancement, edge-enhancement, and noise elimination on entire CXR images by employing histogram equalization (HGE) and unsharp masking edge enhancement. Image normalization and standardization have also been performed on image data. To avoid overfitting L2-weight regularization has been used and gradient-guided class activation maps (GRAD CAM) are used to compute the number of weights of each feature map during training. Softmax class posterior averaging (SCPA) -based ensemble of three CNN models VGG-19, ResNet-18, and
DenseNet-161 have achieved best results, precision, recall, and F1 scores of 0.931, 0.940, and 0.935 respectively.

Rahimzadeh and Attar implemented an ensemble of Xception and ResNet50V2 to detect Covid-19 from the x-ray images. They used two image datasets in their research which have

[2] Joseph's Dataset Source: https://github.com/ieee8023/covid-chestxray-dataset.

[3] RSNA dataset source: https://www.kaggle.com/c/rsna-pneumonia-detection-challenge/data

[4] Dataset source: https://radiopaedia.org/

been retrieved from github[2] and kaggle[3]. ResNet50V2 is slightly different from ResNet-50 around connections between the blocks and ResNet50V2 has good accuracy compared to ResNet50 on ImageNet dataset. Both Xception and ResNet50V2 are generating a 10 × 10 × 2048 feature map in their final layer and ensemble is formed by using these last two layers outputs, concatenate them and then use a convolutional layer which is having a classifier. For training the model, learning rate of 1e-4 is used, the loss function used is categorical cross-entropy. The optimizer used in the network is Nadam. Batch size during training of Xception and ResNet50V2 was equal to 30 and the batch size of 20 is used in the concatenated network. The ensemble model has performed well in detection of Covid-19 from X-ray images with normal accuracy of 91.71% and precision of 92.40%.

Ozcan (2020) used three neural networks GoogleNet, ResNet18 and ResNet50 to do a comparative analysis to determine which neural networks is better in identifying Covid-19 from x-ray images. The x-ray images data was collected from various sources[2,5,6]. For hyperparameter tuning, grid search (GS) is used to tune the parameters for optimization. In GS, the accuracy of each parameter is determined and best set of parameters are selected. GS builds a model of every combination of parameters and uses the combination with best results. The initial learning rate, regularization parameter, momentum and minibatch size used for ResNet50 during training of the model are 0.005, 0.0001, 0.85 and 16 respectively. The ResNet50 has performed better than GoogleNet and Resnet18, with an accuracy of 97.69, the sensitivity, specificity and precision of ResNet50 are also better than GoogleNet and Resnet-18.

Wang and Xia (2018) have created a CNN, ChestNet, to detect the thorax diseases from chest x-ray images. They used the ChestX-ray14 dataset[7] having 112,120 chest x-ray images with 14 labels of various thorax diseases. The ChestNet CNN is having two branches: a classification branch and an attention branch. The classification branch is ResNet-152 CNN trained on ImageNet dataset and the last classification layer, the softmax layer has been replaced with the fully connected layer having 14 neurons, each using the sigmoid activation. The attention branch which is composed of 6 convolutional layers, is connected to the modified ResNet152. The gradient-weighted class activation mapping (Grad-CAM) is used in the attention branch to estimate the class discriminative localization map of each class. For training the network, mini-batch stochastic gradient descent algorithm is used with batch size of 24, learning rate of 0.001, gamma value of 0.1. This CNN network outperformed the ResNet152 and achieved an average accuracy of 78% in classifying the 14 diseases in ChestX-ray14 dataset.

Ansari et al. (2020) used ResNet50 to train a model which would detect pneumonia from chest x-ray images. They used two publicly available datasets for their research, RSNA Dataset[3] and Chest X-Ray Image (CXI) Dataset[8]. The ResNet-50 neural network that is used to train model has pre-trained weights acquired from ImageNet dataset. The optimizer, learning rate and batch size used for training the model on RSNA dataset are ADAM, 0.01 and 20 respectively. The model was trained for 4 epochs only due to already having pre-trained weights, loss function used is Binary cross-entropy. All parameters are same when model is trained on CXI dataset except optimizer which is SGD (Stochastic Gradient

[5] Dataset source: https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia/

[6] COVID-19 database available on: https://www.sirm.org

[7] Chest X-ray 14 dataset: https://arxiv.org/abs/1705.02315

[8] Dataset source: https://data.mendeley.com/datasets/rscbjbr9sj/2

Descent). The accuracy achieved is 97% when RSNA dataset (having almost twice the number of images present in CXI dataset) is used and 94% when CXI dataset is used. The accuracy varies by 0.1% when different training and test ratio are used, maximum accuracy is achieved with 80-20% split.

Wong et al. (2020) have used an ensemble of CNN networks: VGG16+ResNet50 and VGG16+DenseNet121, to classify the chest x-ray images into normal and abnormal category, where abnormal category is having thorax diseases. The datasets used in the research are obtained from MIMIC-4[9], NIH datasets[10]. The CNNs VGG16+ResNet50, pretrained on the ImageNet dataset, are used to extract features from the chest x-ray images and then those features are combined. To reduce overfitting, dilated blocks are used, which are composed of dilated convolutions for multi-scale features, spatial dropout and a skip connection of identity mapping for convergence improvement. Image augmentations is also applied on images (rotation and shifting) to avoid overfitting. Group normalization with ReLU activation is used in training the network and instead of global average pooling, second order pooling is used which maps the features extracted to a higher dimensional space so that they could be more separable. The optimizer used is Nadam with learning rate of 2×10^{-6} and batch size of 48 is used. The epochs are set to 20. Both the ensembles have good accuracy with area under ROC curve of 0.824 and 0.821 for VGG16+DenseNet121 and VGG16+ResNet50 respectively.

Varshini et al. (2019) tested many neural networks Xception, VGG16, ResNet50, DenseNet121 and DenseNet169 to detect pneumonia. They used the pneumonia and normal chest x-ray images obtained from ChestX-ray14 dataset[7]. To remove the gradient vanishing problem which stops the neural network from learning, all the layers are connected with equal feature-sizes directly. The last layer of the neural networks which is the classification layer, is replaced by machine learning classifiers SVM (support vector machines), Naïve Bayes, KNN (K-nearest neighbours), random forest. The hyperparameters are tuned and various combinations of neural networks with machine learning classifiers are tried, SVM (rbf kernel with C=3 and gamma=1.9e-05) has given best results. ResNet50 with SVM as classifier has given good results with area under curve of 0.78 and DenseNet169 with SVM(rbf kernel with C=3.5 and gamma=1.9e-05) has performed best with area under curve of 0.80, compared with all other combinations of neural networks as feature extractors and classifiers SVM, Naïve Bayes, KNN and random forest.

Benbrahim et al. (2020) have used neural networks GoogleNet InceptionV3 and ResNet50 to detect COVID-19 from chest x-ray images. For data processing and storing Apache Spark is used, on which the training of the neural networks is carried out. Pre-trained GoogleNet InceptionV3 and ResNet50 are used in the research. Logistic regression is added as the final layer which is performing the final classification of an image. The image data used in the research study is obtained from Kaggle datasets[11,5]. DeepImageFeaturizer class of sparkdl library of python has been used to implement the logistic regression after the CNN models. Both the neural networks have achieved very good accuracy in classifying the x-ray images, GoogleNet and ResNet's accuracy is 99% and 98% respectively.

[9] Dataset source: https://arxiv.org/abs/1901.07042

[10] Dataset source: https://arxiv.org/abs/1705.02315

[11] Dataset source: https://www.kaggle.com/bachrr/covid-chest-xray

Sethy and Behra (2020) have used multiple CNNs to detect covid-19 from chest x-ray images. The classifier used in the final layer of the neural networks is Support vector machines (SVM). The image data has been retrieved from Github[2] and Kaggle[3]. A total of 11 CNN models (AlexNet, VGG16, VGG19, GoogleNet, ResNet18, ResNet50, ResNet101, InceptionV3, InceptionResNetV2, DenseNet201 and XceptionNet) were trained on the x-ray image data, out of which the combination of ResNet-50 and SVM has achieved highest accuracy and specificity of 95% and 93% respectively, out of all the CNNs models trained on the image data.

Asnaoui et al. (2020) utilized multiple CNNs(VGG16, VGG19, Inception_V3, Xception, DensNet201, MobileNet_V2, Inception_ Resnet_V2) to train models which would detect Pneumonia from chest x-ray image and do a comparative analysis to find out which CNN is giving best results. The image data has been obtained from dataset released by Kermany et al[12]. The dataset is having 4273 x-ray images having pneumonia and 1583 normal x-ray images. Intensity normalization and Contrast Limited Adaptive Histogram Equalization have been applied on input images to improve the visual information of x-ray images. Other data augmentation techniques such as Rescale, image rotation, vertical and horizontal shift, sheer and zoom range have also been applied on image data. Adam optimizer with $\beta 1=0.9$ and $\beta 2=0.999$, learning rate = 0.00001 have used. ResNet has achieved highest accuracy of 97% compared to all other models.

Baltruschat et al. (2019) have used ResNet-50 to train a model which would detect 14 chest diseases such as Pneumonia, Pneumothorax, Effusion, Cardiomegaly, Pleural Thicken etc. The dataset used in the research is the ChestX-ray14 dataset[7], having 112,120 x-ray images with labels: infiltration, effusion, consolidation, pleural thickening, edema, cardiomegaly, fibrosis, atelectasis, emphysema, nodule, hernia mass, pneumothorax, and pneumonia. Two strategies have been adopted in training the ResNet50 model, one with random values as initialization parameters and other with transfer learning using pre-trained weights obtained from training the model on ImageNet dataset. Non-image features such gender, age etc. are also added to the last layer. The trained model with random initialized weights has an average accuracy of 73% without non-image features and 75% with non-image features. The model with pretrained weights has performed better with average accuracy of 82% in both cases, with and without non-image features.

Jain et al. (2020) have employed six neural networks (VGG16, VGG19, ResNet50, Inception-v3 and two models (model 1 and model 2) designed from scratch) to detect pneumonia from chest x-ray images. They have obtained the image data from Kaggle[5], which is having a total 5216 images for training the model and 624 images for testing. The activation functions used in all six models are ReLU and softmax; pooling functions used are average pooling and max-pooling; optimizer used in all models is ADAM. Dropout is used in model 2, VGG16, VGG19, and ResNet to reduce overfitting. VGG16 has outperformed all other models in training with accuracy of 94% and ResNet50 has training accuracy of 94%. VGG19 has best validation accuracy of 88%.

Minaee et al. (2020) have trained four neural networks, ResNet18, ResNet50, SqueezeNet, and DenseNet121, to identify covid-19 from chest x-ray images. The image dataset, which is having around 5000 images, has been prepared by the author who collected x-ray images

[12] Dataset source: https://data.mendeley.com/datasets/rscbjbr9sj/2

from multiple sources and is stored in dropbox[13] whose link could be retrieved from github page of author. Data augmentation such as small rotation, flipping and adding small amount of distortions has been applied on image data. Loss function and optimizer used in all models are cross-entropy and ADAM respectively. The neural networks used in the research have pre-trained weights and have been fine-tuned with 100 epochs. Learning rate is set to 0.0001. The models are evaluated on Sensitivity and Specificity and Sensitivity of all four models are close to 98% and Specificity of SqueezeNet is highest among all four models, which is 93%.

Kassani et al. (2020) have utilized multiple neural networks to train models which would detect COVID-19 from x-ray images. They have used multiple sources to collect image data[2,3,5]. They have tried out various machine learning classifiers such random forest, decision trees etc. after the last layer of neural networks. A comparative study of fifteen neural networks is carried out and on all these fifteen neural networks six machine learning classifiers have been used. Transfer learning is used for all models to overcome over-fitting and get better results. They have utilized image data from multiple sources, named Github and Kaggle. Image normalization and resizing have been implemented on image data as part of data pre-processing. ResNet50 with light GBM has achieved best accuracy of about 98% in all the neural networks.

Lawte et al. (2020) have employed ResNet50 in detection of pneumonia from chest x-ray images. They have used two datasets in their study, RSNA dataset and Chest X-Ray Image (CXI) dataset[12]. They tried other neural networks too like DenseNet-121, Vgg-16 but ResNet-50 when used with transfer learning gave better results. ResNet-50 achieved an accuracy of 97% when trained on RSNA dataset only with the dataset split ratio 80-20; accuracy was 94% when the ResNet-50 was trained on CXI dataset. Therefore, RSNA dataset gives better results for creating a model to diagnose pneumonia.

Shadeed et al. (2020) used Resnet50 for diagnosing thorax diseases. The dataset used in the research is Chest X-ray 14 dataset[7], this dataset is having 112,120 x-ray images with labels: effusion, consolidation, edema, cardiomegaly, atelectasis, emphysema, fibrosis, nodule, hernia mass, infiltration, pneumothorax, pleural thickening, and pneumonia. Resnet overcomes the problem of vanishing gradient, which stops the model from learning when the network is having more than 25 layers. The data pre-processing includes the cropping of rib cage area. For training the model, the mini-batch stochastic gradient descent algorithm has been used with the learning rate of 0.0001. The network is able to achieve an accuracy of 93.03% on training data and 94.49% accuracy on testing data.

Nguyen et al. (2019) have implemented five neural networks (ResNet50, VGG16, VGG19, DenseNet121 and Inception ResNetV2) to detect tuberculosis from chest x-ray images. They have used two datasets in their research, Montgomery and Shenzhen datasets[14] and NIH-14 dataset[7]. They have used transfer learning and used the pre-trained weights, which were trained on ImageNet dataset. They first trained all five neural networks with ImageNet weights and then picked the neural network with best results (DenseNet121 and Inception

[13] Dataset source: https://github.com/shervinmin/DeepCovid/tree/master/data

[14] Dataset source: https://lhncbc.nlm.nih.gov/publication/pub9931

ResNetV2 gave best results) to train on NIH dataset which required to modify the last layer and loss function because NIH dataset has multiple labels. After the model is trained on NIH dataset, it was tested on Montgomery and Shenzhen datasets and the results (0.99 and 0.8 Area under curve for Shenzhen and Montgomery respectively) proved that pre-trained weights from NIH dataset are better than ImageNet dataset pre-trained weights.

3 Research Methodology

3.1 Image classification using CNN

Image classification is a task in which the objective is to output a category to which the image belongs. In binary classification there are only two categories, for example: classifying if the image is of a person or animal is a binary classification. This research is using CNN for binary classification. CNN considers an image as an array of numbers. This array of numbers has high values where intensity of colour is high and low-numbered values where intensity of colour is low.

CNNs are very popular when it comes to image classification. Due to the dense architectures of CNN, the hidden layers of CNN are able to identify many features of input images which uniquely identifies an image, the various shapes and patterns present in an image are captured by CNN hidden layers which helps to classify an image.

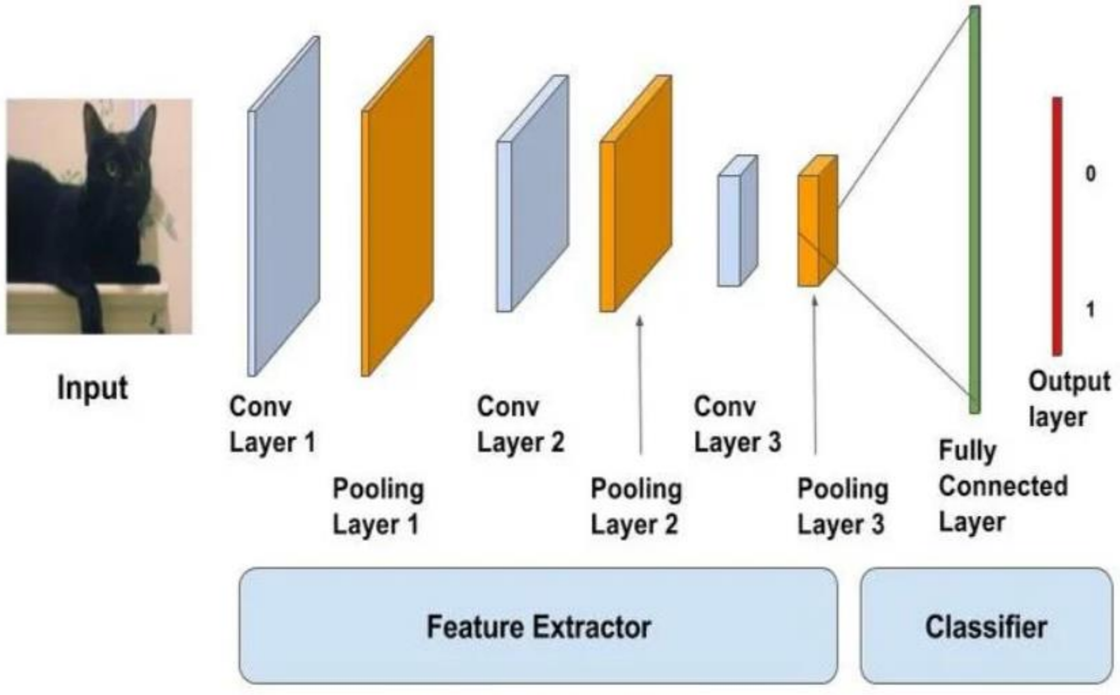


Fig: 2 - CNN design

[15]

[15] Fig 2: Source: https://www.learnopencv.com/image-classification-using-convolutional-neural-networks-in-keras/

CNN has various layers, convolution layer is the first layer of CNN, in this layer when an input image is passed to the network, a filter, which is an array of numbers of size much smaller than the size of the image (for example: 3x3), passes across the entire image array and outputs a result.

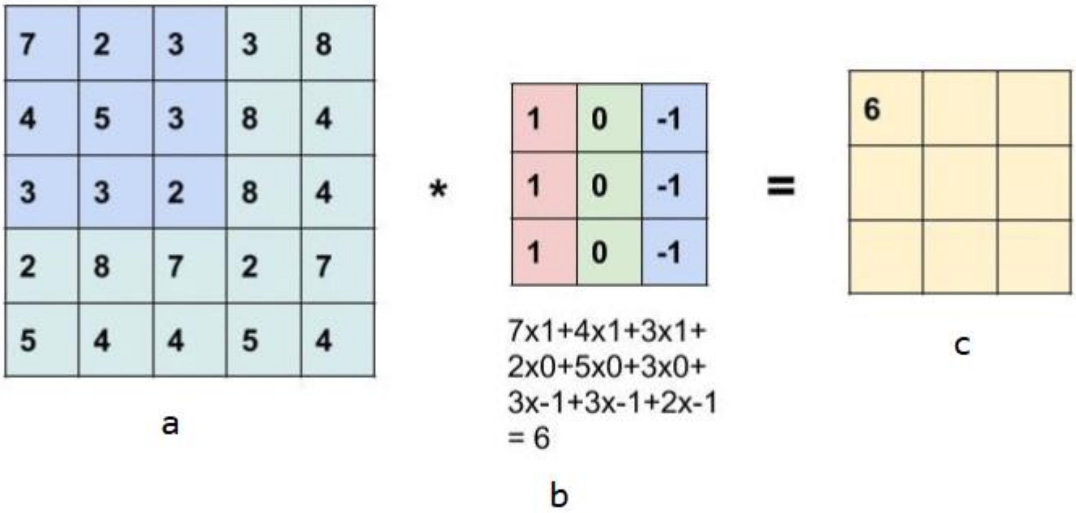


Fig: 3 - a:Image array, b:Filter, c:Result

[16]

For example, the filter would start from the top left and all the image array numbers would be multiplied by filter numbers and then added to get one number output from one location where filter is present. After this the filter would move across right by number of units mentioned in the parameter stride, if the stride is of “2”, then filter would move two places right and perform the operation of multiplication and addition. This way the filter moves across the entire image. At every layer, the filter identifies a pattern which helps in figuring out the category of the image. The below image is an example how in one layer different shapes of the image are considered and in another layer other shapes of the image are focussed. At the last layer, the fully connected layer, vector of all categories of the image is the output which is the final result. For example, if there are four categories into which the image could belong to, then the vector could be having values [0.15, 0.19, 0.11, 0.55], the result would be that the image belongs to the fourth category as the probability for the fourth category is the highest.

3.2 CNN parameters and backpropagation

[16] Fig 3: Source: https://www.learnopencv.com/image-classification-using-convolutional-neural-networks-in-keras/

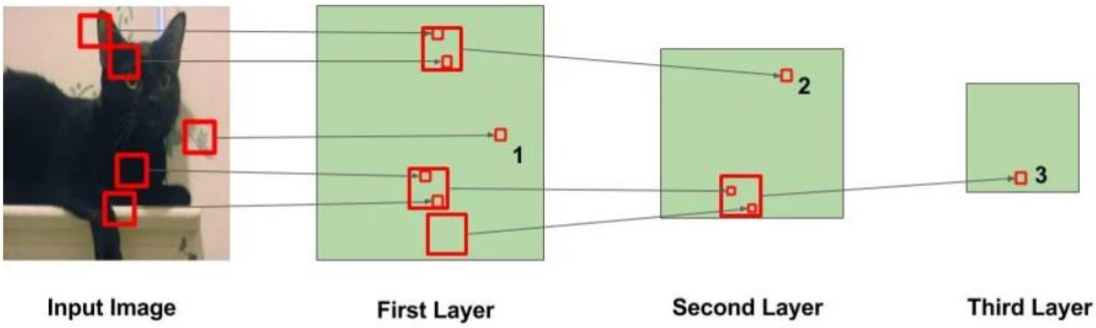
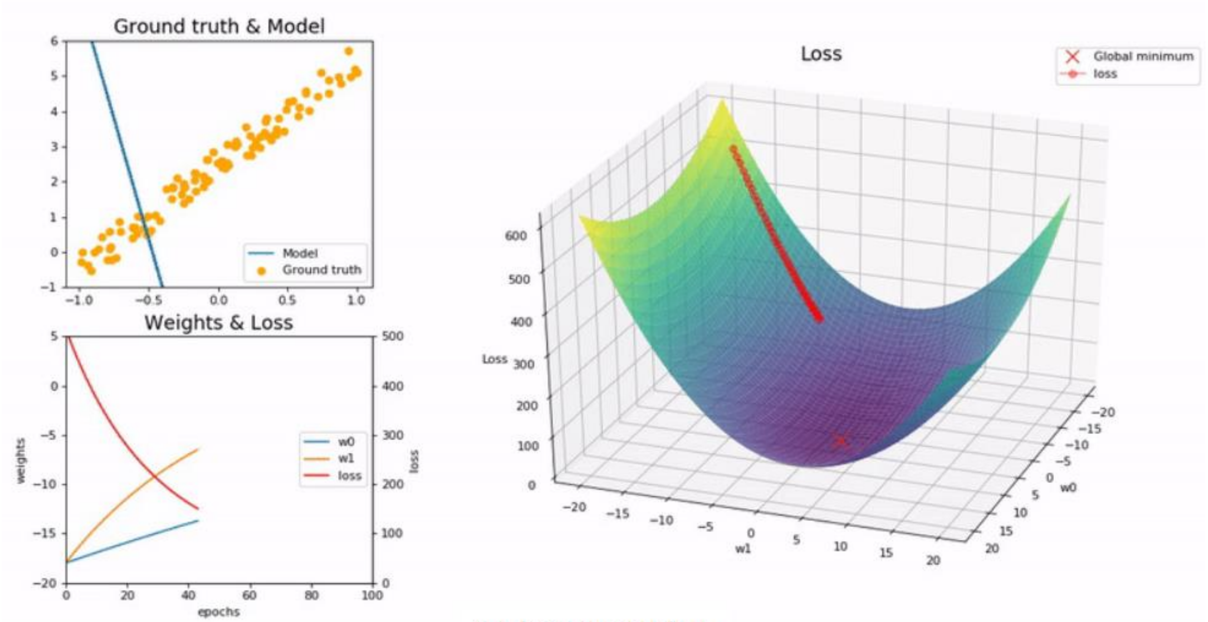


Fig: 4 - Fifferent features identified by filters across various layers

[17]

Backpropagation is a very important aspect of CNN. Through backpropagation the filter values are updated at each iteration with the help of loss function and optimizer. CNN has random values of image filters in the beginning(except in transfer learning where weights are loaded of pre-trained model), so the features which should be extracted to uniquely identify an image, may not be identified because of the filter values. When an image passes through the convolutional layers in the beginning during training, the neural network may classify the image to incorrect category but CNN already knows what is the correct category as during training the correct labels are provided. After classifying the image into incorrect category, the loss function calculates how much the calculations have been deviated because of which the classification of image is incorrect. Then the weights are adjusted as per the learning rate mentioned in the network, if the learning rate is high then there would be big adjustments in the weights of the network which could make the model less precise.



Backpropagation

[17] Fig 4: Source: https://www.learnopencv.com/image-classification-using-convolutional-neural-networks-in-keras/

The other parameters which are important in understanding the working of the neural network are pooling function, activation function, filter size, zero padding, stride, dropout, optimizer. Pooling function is responsible for reducing the output obtained from one convolution layer to reduce the computations for further networks and also reduce the parameters. Two popular pooling functions are Maxpooling and Average pooling.

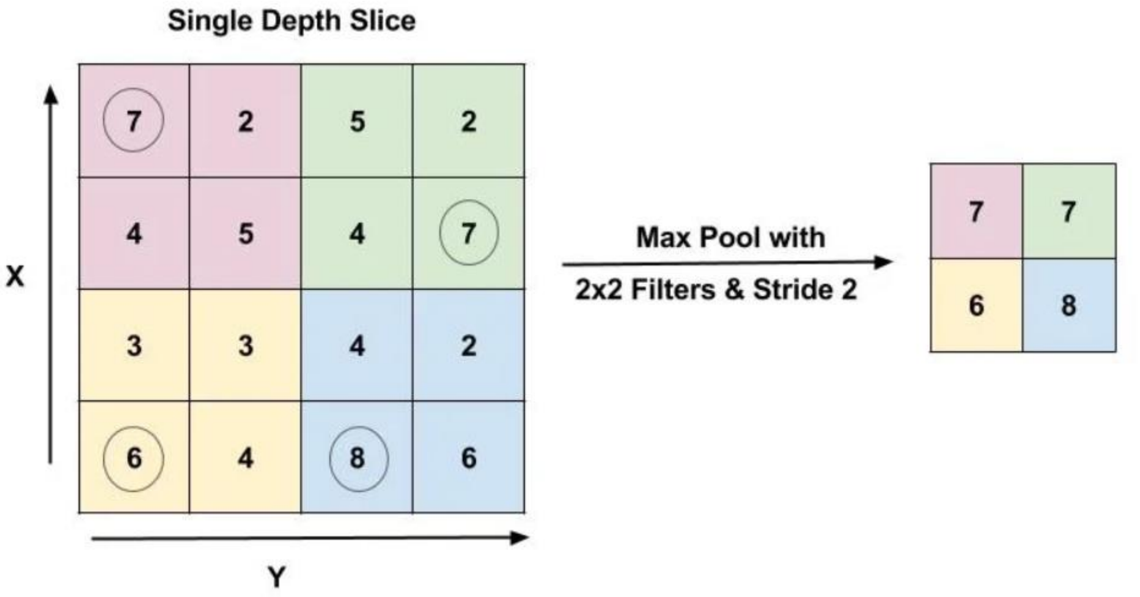


Fig: 5 - Maxpooling

[18]

The activation function is responsible for getting the final output from one convolutional layer after the multiplication and addition of image array with filter, non-linearity in the training process is also handled by activation function. Activation function decides whether the neuron will fire or not. Filter size is an NxN array of numbers which convolves around the entire image to extract the features from the image in convolution layer. Stride is the number which decides how much the filter would shift after multiplication between image array and filter, if the stride is of 2 units, then the filter would shift by 2 units from its current location.

[18] Fig 5: Source: https://www.learnopencv.com/image-classification-using-convolutional-neural-networks-in-keras/

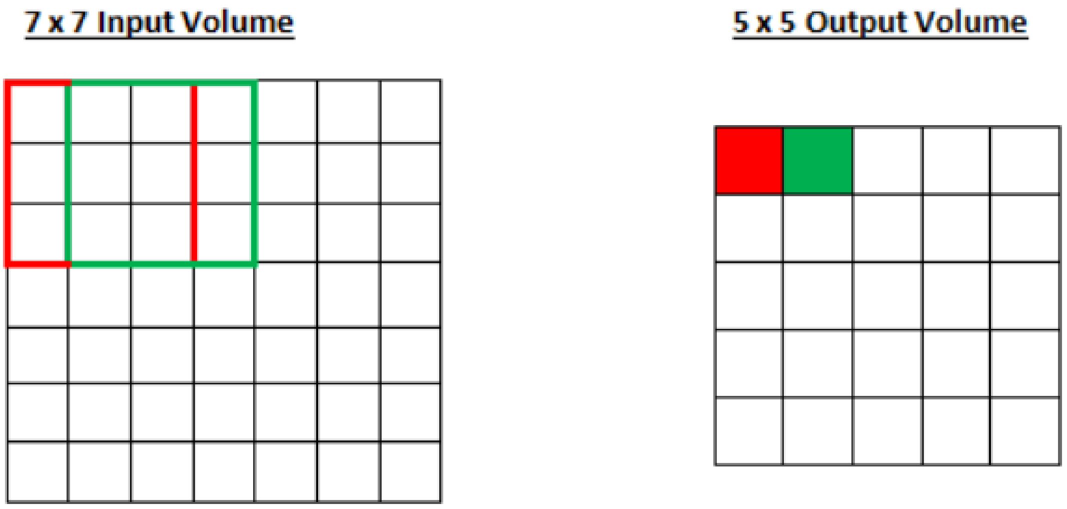
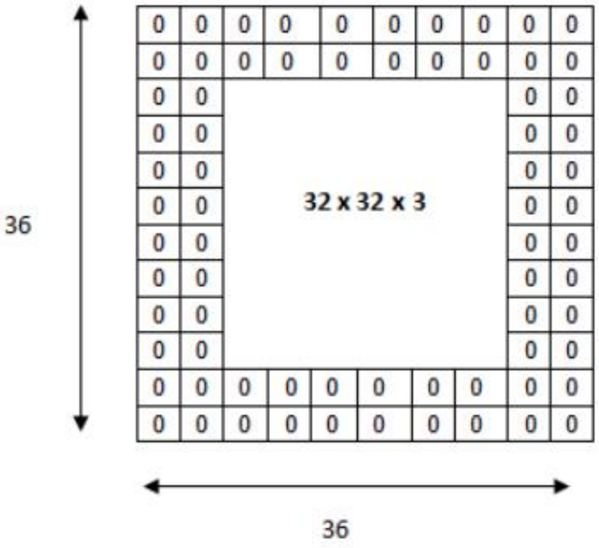


Fig: 6 - Stride

[19]

Zero padding is the addition of elements having value zero to the output of convolution layer to maintain the size of the output so that the array size is same before and after the convolution layer.



The input volume is 32 x 32 x 3. If we imagine two borders of zeros around the volume, this gives us a 36 x 36 x 3 volume. Then, when we apply our conv layer with our three 5 x 5 x 3 filters and a stride of 1, then we will also get a 32 x 32 x3 output volume.

Fig: 7 - Zeropadding

[20]

[19] Fig 6, 7: Source: https://deepai.org/machine-learning-glossary-and-terms/stride

Dropout is used to randomly drop some layers from the neural network during training to avoid overfitting. If dropout of 0.3 is used, then it would mean that 30% layers will be dropped.

3.3 Dataset Description

For any research to be successful, the data plays a very important role. For neural networks to perform efficiently, huge amount of data is required. Mild Pneumothorax is very hard to detect and to train a neural network to determine if the x-ray image has pneumothorax or not, thousands of x-ray images having mild pneumothorax are required.

The dataset used in this research has been obtained from Kaggle competition hosted by Society for Imaging Informatics in Medicine (SIIM) and American College of Radiology (ACR). Along with SIIM and ACR, Society of Thoracic Radiology (STR) and MD.ai have also helped in annotating the dataset. The dataset is having a total of 12,047 images, in which 10,675 images will be used to train the neural networks and 1,372 images will be used to test the model. Out of the 10,675 images used for training 2,379 images are having pneumothorax and rest of the 8,296 images do not have pneumothorax. Out of 1,372 images which will be used to test the model, 290 images have pneumothorax and 1,082 do not have pneumothorax. The images are named in a way that from the name of the image it could be figured out if the image is having pneumothorax or not, for example: image named ‘19_train_1_’, means that the image is to be used to train the image and is having label ‘1’ which implies that the image is having pneumothorax; image named ‘22_test_0_’, means that the image is to be used to test the image and is having label ‘0’ which implies that the image is not having pneumothorax. Apart from the names of the images an excel sheet is also having details of images, the excel sheet is having three columns named 'new_filename', 'ImageId' and 'has_pneumo'. Column 'new_filename' is the name of the file which is explained in the previous two examples; column 'ImageId' is a long list of numbers, for example: ‘1.2.276.0.7230010.3.1.4.8323329.4904.1517875185.355709’, this column will not be used in the research and will be ignored; column 'has_pneumo' is having two values 0 and 1, each image is having either 0 or 1, where 0 implies that image is not having pneumothorax and 1 implies that image is having pneumothorax. This excel sheet will be imported in python to create a numpy array having labels for images, from this array the model will find out if the image which is input to the model to train the model is in category of images having pneumothorax or in another category.

This dataset of images is not violating any data governance ethics. The images present in the dataset are not having any information which would uniquely identify a person, therefore, the person’s right to privacy is protected. The dataset is also an open dataset which is publicly available, so the creators of the dataset have publicly uploaded the dataset for the researchers to use for medical purposes. The excel sheet which is having the labels from which it could be found out if the person is having pneumothorax or not is also not having any details of a person through which a person could be uniquely identified. No such details like person’s address, phone number, location, name are present in the excel sheet and image dataset.

4 ResNet50 Design Specification

ResNet50 is a convolution neural network with 50 layers. ResNet-0 is a very powerful neural network which has achieved very good results in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) competition of year 2015. ILSVRC is an annual competition is which the objective to design a neural network which could classify images of dataset ImageNet. ImageNet is a very large dataset having more than fourteen million images and 20,000 categories. Many industry leaders and researchers participate in the competition to develop a neural network which would achieve highest accuracy. ResNet was the winner of ILSVRC 2015 competition. ResNet also won the first place in ImageNet localization, COCO 2015 competition. COCO is also a very large dataset.

There are many variants of ResNet neural network. Some of them are: ResNet18, ResNet34, ResNet50, ResNet101, ResNet110, ResNet152, ResNet164, ResNet1202 etc. ResNet50 is placed between large neural networks with more than 100 layers and small neural networks having about 20 layers. Large neural networks face the problem of gradient vanishing which stops their learning process after some time during training which is discussed later. The ResNet-50 is the network which is placed between the large neural networks and small neural networks.

The main aspect of this CNN is the introduction of the residual function. The main objective of the residual function is to tackle the problem of gradient vanishing. Gradient vanishing is the problem faced by neural networks which have a lot of layers and are very deep. When the network has a lot of layers the accuracy of the training of model remains same or starts decreasing due to the vanishing gradient effect. When the error is being calculated during the backpropagation process, the gradient values are calculated which would update the weights of the network, but the gradient values becomes smaller and are unable to reach till the bottom of the network which causes the initial layers of the network to stop updating their weights which results in diverging the learning of the network and hence the accuracy will remains same or starts to decrease. Below mentioned image explains the problem of vanishing gradient.

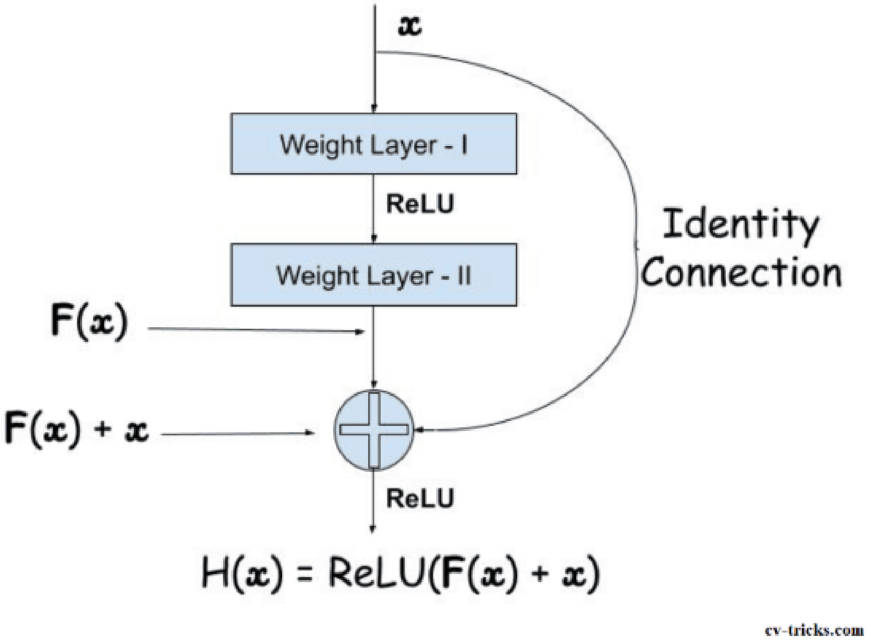


Fig: 8 - A residual block of deep residual network

[21]

The above picture explains the working of the most important modification which makes ResNet stands out, the presence of skip connection, identity mapping. This function identity mapping does not have any parameters associated to it, its only objective is to add the output from the previous layer to the current layer. But if the two layers whose output is to be added is having different dimensions, then they won't add up, to overcome this problem a separate parameter, W, is added to make the dimensions same. It is better for the layer to consider the previous layer output and learns from it to improve the overall accuracy, therefore, the previous layer output plays an important role in the training process in ResNet50 to improve the accuracy and avoid the vanishing gradient problem.

ResNet50 design can be divided into four stages. In stage one, the image is passed having dimensions 224x224x3, then convolutions and maxpooling is performed on the image array using 7x7 and 3x3 kernel sizes. After this step every stage of the network will have three or more residual blocks and each block is having three or more convolution layers. In the image below, the arrow indicates where the residual function will be applied, after three convolutions. The dashed curved arrow represents that the convolution is performed with stride of two, to reduce the size of output by half. Similar operations are performed in different stages of ResNet50 and at the end of the CNN is the fully connected layer which has a total number of outputs equal to the number of categories in which image can be classified.

[21] Fig 8: Source: https://cv-tricks.com/keras/understand-implement-resnets/

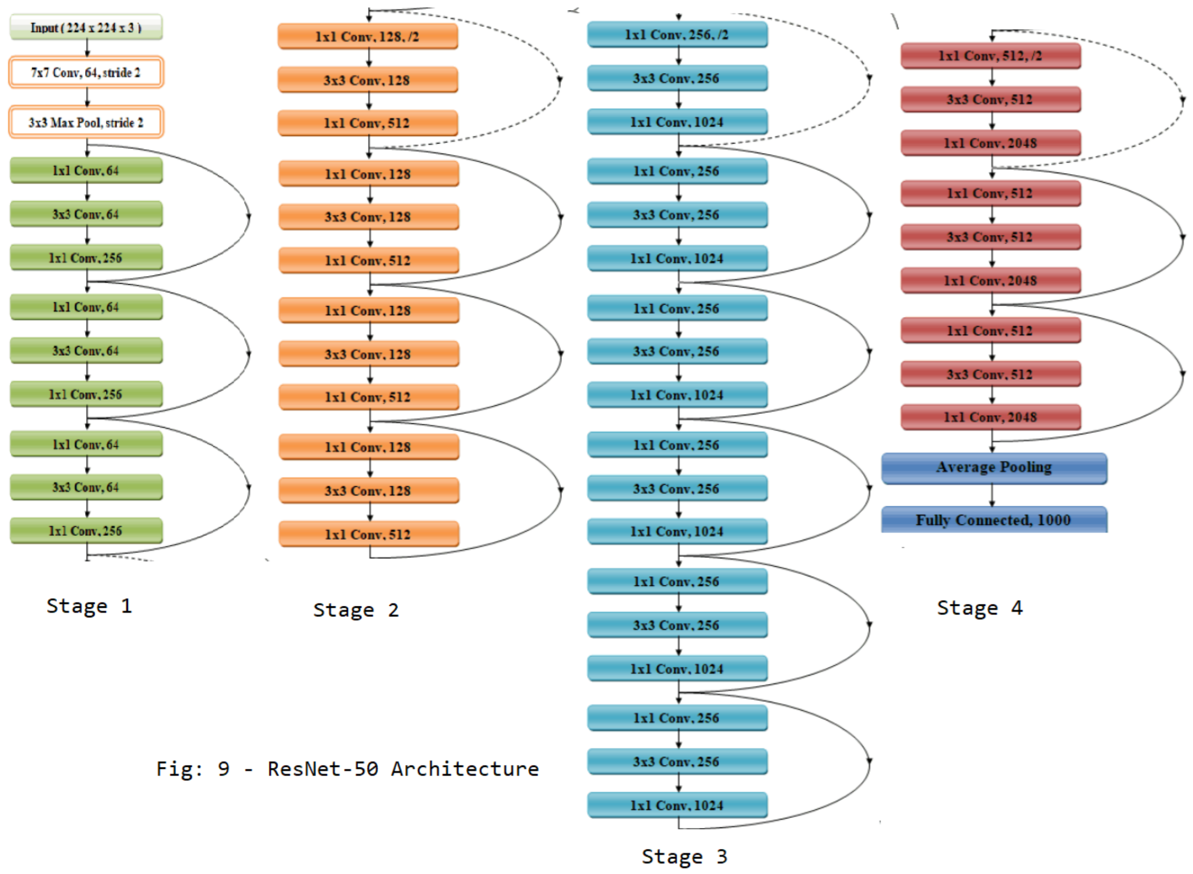


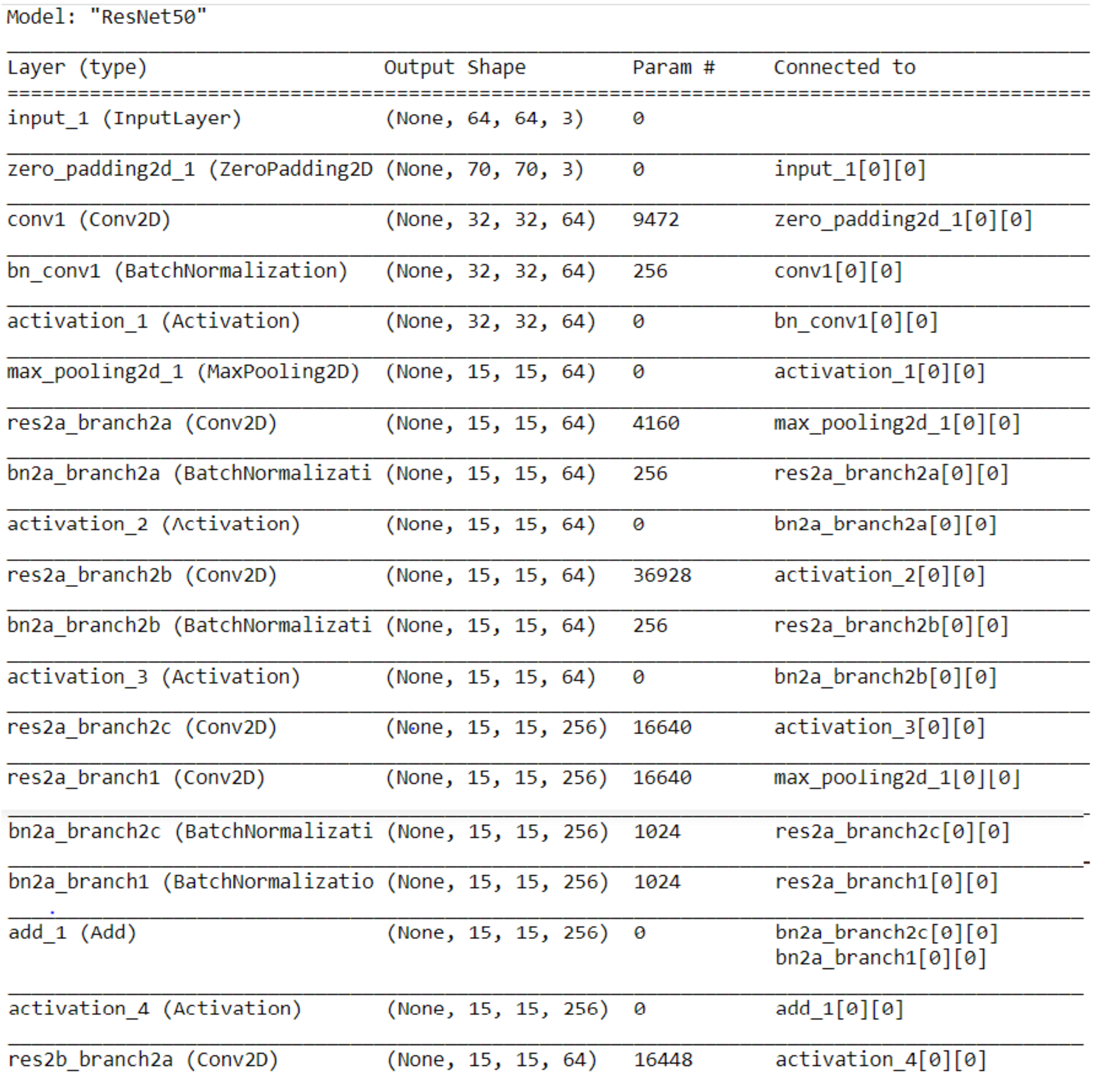
Fig: 9 - ResNet-50 Architecture

[22]

The summary function is used to find information about the layers present in the network and their order. The shape of the output from the layer can also be found out by the summary function. The number of parameters, which are also called weights can also be found out with the help of summary function. The total number of parameters of the entire model can also be found by using the summary function. The model is having a total number of 23,591,810 parameters out of which the trainable parameters are 23,538,690 and 53,120 parameters are non-trainable. The trainable parameters are the ones whose values gets updated during the backpropagation when the network is training. The summary function outputs a textual detailed summary of the entire model. A summary of the current ResNet50 model which is used in this research is presented below. This summary is not the entire summary because the image would be very long and will take a lot of space, therefore image is cropped and the top part of the image is presented below.

[22] Fig 9: Source: https://cv-tricks.com/keras/understand-implement-resnets/

[23]



```
Model: "ResNet50"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_1 (InputLayer)            (None, 64, 64, 3)    0
__________________________________________________________________________________________________
zero_padding2d_1 (ZeroPadding2D (None, 70, 70, 3)    0           input_1[0][0]
__________________________________________________________________________________________________
conv1 (Conv2D)                  (None, 32, 32, 64)   9472        zero_padding2d_1[0][0]
__________________________________________________________________________________________________
bn_conv1 (BatchNormalization)   (None, 32, 32, 64)   256         conv1[0][0]
__________________________________________________________________________________________________
activation_1 (Activation)       (None, 32, 32, 64)   0           bn_conv1[0][0]
__________________________________________________________________________________________________
max_pooling2d_1 (MaxPooling2D)  (None, 15, 15, 64)   0           activation_1[0][0]
__________________________________________________________________________________________________
res2a_branch2a (Conv2D)         (None, 15, 15, 64)   4160        max_pooling2d_1[0][0]
__________________________________________________________________________________________________
bn2a_branch2a (BatchNormalizati (None, 15, 15, 64)   256         res2a_branch2a[0][0]
__________________________________________________________________________________________________
activation_2 (Activation)       (None, 15, 15, 64)   0           bn2a_branch2a[0][0]
__________________________________________________________________________________________________
res2a_branch2b (Conv2D)         (None, 15, 15, 64)   36928       activation_2[0][0]
__________________________________________________________________________________________________
bn2a_branch2b (BatchNormalizati (None, 15, 15, 64)   256         res2a_branch2b[0][0]
__________________________________________________________________________________________________
activation_3 (Activation)       (None, 15, 15, 64)   0           bn2a_branch2b[0][0]
__________________________________________________________________________________________________
res2a_branch2c (Conv2D)         (None, 15, 15, 256)  16640       activation_3[0][0]
__________________________________________________________________________________________________
res2a_branch1 (Conv2D)          (None, 15, 15, 256)  16640       max_pooling2d_1[0][0]
__________________________________________________________________________________________________
bn2a_branch2c (BatchNormalizati (None, 15, 15, 256)  1024        res2a_branch2c[0][0]
__________________________________________________________________________________________________
bn2a_branch1 (BatchNormalizatio (None, 15, 15, 256)  1024        res2a_branch1[0][0]
__________________________________________________________________________________________________
add_1 (Add)                     (None, 15, 15, 256)  0           bn2a_branch2c[0][0]
                                                                 bn2a_branch1[0][0]
__________________________________________________________________________________________________
activation_4 (Activation)       (None, 15, 15, 256)  0           add_1[0][0]
__________________________________________________________________________________________________
res2b_branch2a (Conv2D)         (None, 15, 15, 64)   16448       activation_4[0][0]
.......................
```

Fig: 10 - ResNet-50 model summary

5 Implementation

5.1 Dataset pre-processing

The x-ray images are concentrated on the thorax part of the body. This part is only required to determine if the image is having pneumothorax or not. There is no other part of the body in the x-ray images, therefore there is no need to crop the images, whole image will be used as input to train the neural network. Even if the images are cropped from one part then other image would be having important aspect in that part, so cropping of images has been ruled out. The dimensions of the images are 1024x1024, this needs to be changed into 224x224 as images of this dimensions only can be used as input to train the neural networks ResNet50.

[23] Fig 10: This summary is the screenshot of the result of python code used in the research

To change the dimensions of the images CV2 library is used. Numpy of python is used to get all images into one array and also normalize all images so all images would have values of pixels ranging from zero to one. CV2 library is also used to import the images in python. Keras library function np_utils will be used to change the labels of the images into binary form of zeros and ones. Python library pandas is used to extract the labels of the images from the excel sheet.

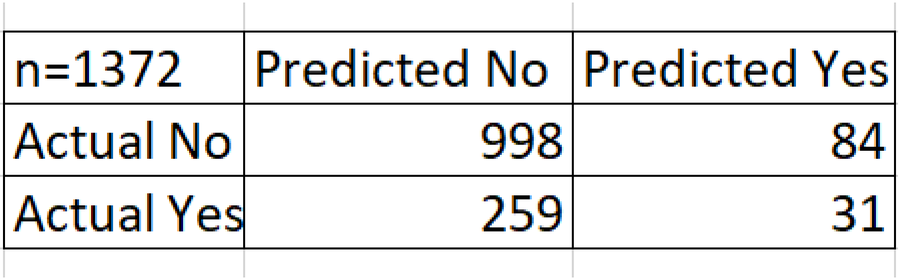
5.2 Model Training

The model is designed using python library keras. The training part of the model was carried on personal laptop of author which is having configuration as mentioned: Model - HP Pavilion Laptop 15-cw1xxx, RAM 16 GB and processor AMD Ryzen 7 3700U with inbuilt Radeon Vega Mobile Gfx having frequency 2.3 GHz. The training part was also tried on Google Colab but the pre-processing part of the image data was giving error of insufficient RAM as there are more than 10,000 images. This issue was handled on the personal laptop by importing the images one by one and after one image is imported, the pre-processing part was carried out right after that and then the image is added to one array which would have all images. This process saved a lot of RAM as there would only be one variable used to store the pre-processed images and the variables which are used for pre-processing part are destroyed right after one image is imported and pre-processed.

After all the images have been imported, the model was designed using Keras library and compiled. The optimizer used in the training of the model is ADAM. This optimizer is better than SGD(Stochastic gradient descent) optimizer. The loss function used is binary_crossentropy as the classification is of binary type and the output that is received is a probability value between zero and one. The batch size is taken as 256 and various epoch sizes are used to train the model multiple times to find out the best configuration. Model training when epochs are 15 gave better results when epochs were 25 and 35.

6 Evaluation

After training the model, the model was tested on test images and the result was captured in an array and average accuracy achieved was 74%. The actual result of the images from the excel sheet was in the form of 0 and 1 where 0 means x-ray is not having pneumothorax and 1 means the image is diagnosed with pneumothorax. But the predicted values are in the binary form where [0. 1.] is equivalent to 1 implying the image is having pneumothorax; [1. 0.] is equivalent to 0 meaning that image is not having pneumothorax. So, the result array of numbers has to be converted back into non-binary form. Then confusion_matrix of sklearn library is used to calculate the confusion matrix and then use the values from confusion matrix to calculate Sensitivity and Specificity.



n=1372	Predicted No	Predicted Yes
Actual No	998	84
Actual Yes	259	31

Fig: 11 - Confusion Matrix [24]

The specificity and sensitivity calculated are 92% and 10% respectively.

7 Conclusion and Future Work

The high specificity implies that x-ray images which are actually negative meaning which are not having pneumothorax are predicted correct with 92% probability and in 8% cases model is making incorrect prediction. Less Sensitivity implies that model is not very good predicting pneumothorax.

Different optimizers like Adamax or Nadam could be tried to train the model and get good accuracy. The updated version of ResNet50 which is ResNet50 V2 could be used as the new version has achieved better accuracy than ResNet50. The pneumothorax which is not severe is very difficult to detect even by radiologist as the features which are difficult to observe from naked eye, a neural network which is much dense than ResNet50 like ResNet152 could be tried to check the accuracy of the network. Machine learning classifiers like SVM (Support vector machines), decision trees, logarithmic function could be used as final classifier as many researchers have used them and got better results.

References

Karim, M., Döhmen, T., Schuhman, D., Decker, S., Cochez, M. and Beyan, O., 2020. Deepcovidexplainer: Explainable COVID-19 Predictions Based On Chest X-Ray Images, Preprint. [online].

Rahimzadeh, M. and Attar, A., 2020. A Modified Deep Convolutional Neural Network For Detecting COVID-19 And Pneumonia From Chest X-Ray Images Based On The

[24] Fig 11: This is the screenshot of the result of python code used in the research

Concatenation Of Xception And Resnet50v2, Volume 19, 2020, 100360. [online] sciencedirect.

Ozcan, T., 2020. A Deep Learning Framework For Coronavirus Disease (COVID-19) Detection In X-Ray Images, Preprint. [online] Assets.researchsquare.com.

Wang, H. and Xia, Y., 2018. Chestnet: A Deep Neural Network For Classification Of Thoracic Diseases On Chest Radiography, Volume: 1807.03058, 2018 [online] Arxiv.org.

Ansari, N., Faizabadi, A., Motakabber, S. and Ibrahimy, M., 2020. Effective Pneumonia Detection Using Resnet Based Transfer Learning. Volume: 82. 15146 - 15153, 2020[online] researchgate.

Wong, K., Moradi, M., Wu, J., Pillai, A., Sharma, A., Gur, Y., Ahmad, H., Chowdary, M., Chiranjeevi, J., Polaka, K., Wunnava, V., Reddy, D. and Mahmood, T., 2020. A Robust Network Architecture To Detect Normal Chest X-Ray Radiographs - IEEE Conference Publication. pp. 1851-1855, doi: 10.1109/ISBI45749.2020.9098671[online]

Varshni, D., Thakral, K., Agarwal, L., Nijhawan, R. and Mittal, A., 2019. Pneumonia Detection Using CNN Based Feature Extraction - IEEE Conference Publication. 2019, pp. 1-7, doi: 10.1109/ICECCT.2019.8869364.[online] Ieeexplore.ieee.org.

Benbrahim, H., Hachimi, H. and Amine, A., 2020. Deep Transfer Learning With Apache Spark To Detect COVID-19 In Chest X-Ray Images. Volume 23, Number S, pp. S117–S129, 2020 [online] Romjist.ro.

Sethy, P. and Behera, S., 2020. Detection Of Coronavirus Disease (COVID-19) Based On Deep Features, DOI: 10.20944/preprints 202003.0300.v1 [online] researchgate.

Asnaoui, K., Idri, A. and Chawki, Y., 2020. Automated Methods For Detection And Classification Pneumonia Based On X-Ray Images Using Deep Learning. Volume: 2003.14363, 2020 [online] Arxiv.org.

Baltruschat, I., Saalbach, A., Nickisch, H., Grass, M. and Knopp, T., 2019. Comparison Of Deep Learning Approaches For Multi-Label Chest X-Ray Classification, Scientific Reports. 10.1038/s41598-019-42294-8. [online]

Jain, R., Nagratha, P., Katariaa, G., Kaushika, S. and Hemanth, J., 2020. Pneumonia Detection In Chest X-Ray Images Using Convolutional Neural Networks And Transfer Learning, Volume 165, 1 December 2020, 108046 [online] sciencedirect.com.

Minaee, S., Yazdani, S., Sonka, M., Kafieh, R. and Soufie, G., 2020. Deep-COVID: Predicting COVID-19 From Chest X-Ray Images Using Deep Transfer Learning. Volume 65, October 2020, 101794. [online] sciencedirect.com.

Kassani, S., Kassasni, P., Schneider, K. and Wesolowski, M., 2020. Automatic Detection Of Coronavirus Disease (COVID-19) In X-Ray And CT Images: A Machine Learning-Based Approach. Preprint.[online] researchgate.

Lawte, S., Sharma, K. and Sardar, T., 2020. Review On Pneumonia Detection From Chest X-Ray Using Deep Learning Approach. Volume 8 Issue VI June 2020, ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429 [online] ijraset.

Shadeed, G., Tawfeeq, M. and Mahmoud, S., 2020. Deep Learning Model For Thorax Diseases Detection. 18. 441. 10.12928/telkomnika.v18i1.12997. [online] researchgate.

Nguyen, Q., Nguyen, B., Dao, S., Unnikrishnan, B., Dhingra, R., Ravichandran, S., Satpathy, S., Raja, P. and Chua, M.. Deep Learning Models For Tuberculosis Detection From Chest X-Ray Images, 2019, pp. 381-385, doi: 10.1109/ICT.2019.8798798. [online] ieeexplore.