

Lung Pathology Detection Using Deep Learning

Ankith Kumaran R, Rohith Kumar B, Dr. J. Kalaivani
SRM Institute of Science and Technology,
Kattankulathur

Abstract:- Since the advent of COVID-19, there has been a weakness to the lungs due to infections and the immunity of the same is more so ever under threat; the coronavirus disease 2019 (COVID-19) that began at the start of this decade has had deleterious effects on the world and is still raging on, causing substantial losses to people's lives. Consequently, there has also been an increase in lung related cases, since COVID-19 leads to weakening of lungs. This needs to be addressed immediately, or more lives will be lost to this deadly disease. With the advent of Artificial Intelligence and Deep Learning, it has provided hope for fast and accurate medical diagnosis, which is vital for healthcare facilities. Image Classification is the technique of extracting patterns and information from an image that plays a major role in medical image classification. With the applications of image classification and deep learning, the disease pneumonia can be identified just by feeding images to the processing model, which will then classify it into distinct classes - primarily affected or normal.

Keywords:- Pathology, Diagnosis, Deep Learning, Classification

1 Introduction

Ailments affecting the lungs and respiratory system are most prevalent in three regions: The lung airway, the tissues of the lung or the blood vessels surrounding the lung. These maladies can be fatal if not diagnosed at the earliest possible and remedial measures administered. To counter this fatal observation, prompt diagnosis of these diseases are vital, and the need of the hour considering the ongoing global pandemic resulting in weakening of respiratory immunity. Assistance to the medical community

can present itself in the form of diagnosis systems engineered using the advancements in the field of image classification and deep learning. Chest radiographs, made available similar to the set of images used by this project published by the National Institutes of Health (NIH), can aid and assist in the creation of these medical systems and help elevate accuracy of diagnosis, which is essential in diagnosis systems.

The objective of this paper is to develop and compare CNN models [1] considering previous attempts [2][3][4][5] to create models, and address the drawbacks of the past attempts in order to maximise efficiency, and perform a comparative analysis. To achieve the highest accuracy, the models trained in this project are MobileNet[7], DenseNet[8] and ResNet[9] each separately concatenated with a GAP Layer[20], a subset of Convolutional Neural Networks designed for embedded vision applications [15] with five convolution layers, and a fully connected layer as the output and final layer. The numbers of layers were decided through extensive training to find the optimal accuracy taking into account the computational costs and complexity associated with it. To achieve the optimal accuracy, three CNN models [13] are created which are then trained to classify and hence be able to accurately predict radiograph images as normal (or) affected. Accuracy was determined using evaluation metrics such as ROC curves[24] and similarity matrices, which are favored in medical imaging systems due to its ability to depict how accurately the model is able to classify the radiographs.

Related Work

There have been previous attempts to tackle this problem of classifying radiographs with near

perfect accuracy. Listed below are some citations which we used to infer our study:

Xiaosong Wang et al. [2] published their paper which aimed at collecting chest radiographs from different sources using natural language processing extraction methods and subsequently named the dataset 'ChestX-ray8' comprising of 108,948 images segregated into eight disease classes. The paper was also able to demonstrate a vanilla CNN model which was able to detect and locate thoracic diseases of these eight classes spatially. They implemented a supervised multi-label image classification and localization framework, which was validated using the same dataset. As a result, the proposed model was able to achieve an accuracy of 0.7891 on average across the eight diseases.

Gerardo Luis et al. [3] proposed their paper, the objective being a lung disease detection model implementing feature extraction and extreme machine learning, to primarily detect three prominent diseases; Pulmonary Fibrosis(PF), Chronic Obstructive Pulmonary Disease(COPD), and Healthy Lung (HL) samples. The dataset used for the model was an image cluster set produced with the assistance of the Walter Cantidio Hospital, Fortaleza. The proposed methodology extracted features from segmented CT images and applied extreme machine learning(ELM) to classify the lung disease in three classes. Additionally, this paper also put forward an algorithm to analysis of CT images on personal workstations, which reduced software dependencies and cost incurred with CT workstations. The model's performance was evaluated using metrics and confusion matrices, resulting in an 0.85 due to sensitivity and specificity issues.

Jooae Choe et al. [4] approached the problem statement by applying content-based image retrieval using deep learning for detecting interstitial lung disease using CT radiographs. The paper was based on applying content based image retrieval on a database consisting of four disease classes - usual interstitial pneumonia (UIP)

, nonspecific interstitial pneumonia (NIP), cryptogenic organizing pneumonia and chronic hypersensitivity pneumonia. Natural Language queries were executed to extract images from the resulting database. A deep learning model was created and the ability of the model to compare and contrast the extent of different regional based diseases were quantified using evaluation metrics. The results were compared before and after content-based image retrieval and it yielded positive results, with the overall accuracy increasing from 0.524 to 0.728.

Naman Gupta et al. [5] published their findings of applying evolutionary algorithms to perform the same task. Four evolutionary algorithms were applied to accurately classify images into two classes; Chronic Obstructive Pulmonary Disease(COPD) and Fibrosis. These evolutionary algorithms ensure that the optimal subset was created for model training with lower dimensions to reduce computational costs. These optimal subsets were then fed into different classification models such as random forest classifier, support vector machines and so on. The highest accuracy was computed at 0.994 by K-nearest neighbors.

Siddanth Tripathi et al. [6] used a publicly published dataset named 'ChestX-ray 8' to train different version of deep learning models and compared their accuracy dependent on certain factors. Machine Learning as well as Deep Learning techniques were deployed in this paper, primarily CNN and CapsNet using spatial transformation techniques. Training and evaluation resulted in an accuracy of 0.67 with a combination of CNN built upon visual geometry group and a spatial transformer network.

3 Proposed Work

Pre-trained convolutional neural networks(CNN) namely MobileNet[7], DenseNet[8] and ResNet[9] are going to function as the base layer of our model and transfer learning[25] methods are applied to fine tune the network to this particular dataset. In addition, four supplementary layers consisting of GlobalAveragePooling [20], dense and dropout layers are also added to maximise accuracy and efficiency. Past attempts have only used pre-trained models but in this paper, with the applications of transfer learning and additional layers, a higher accuracy will be reached with insignificant computational costs. The final output will have the same amount of neurons as the number of classes(14 disease classes + 1 normal class) and a sigmoid activation will be employed. After the model is trained, the network will be able to output percentages corresponding to which pathology the radiograph most closely exhibits, as most of these diseases have similar symptoms and conditions. Thus, our model can be used as a preliminary scanning tool for medical practitioners to streamline their process of disease evaluation and elimination. These models will then be evaluated based on metrics such as categorical accuracy, ROC and AUC curves.

The dataset we will be training our model on,

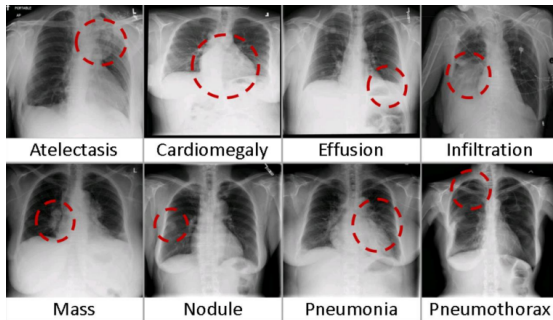


Figure 1: Pathology Description

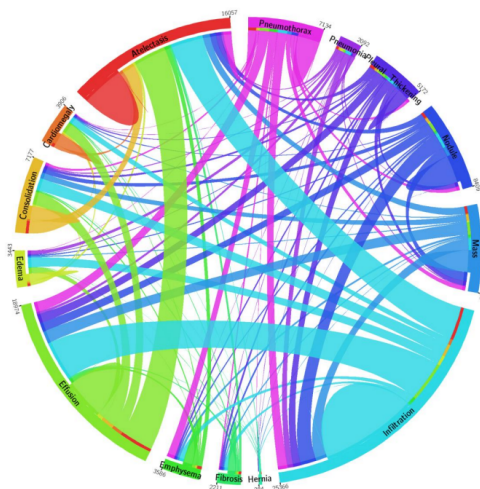


Figure 2: Pathology Correlation

titled ChestX-ray8[2] was released by the National Institute of Health, the central governing agency in The United States of America and consists of more 110,000 chest radiographs with labelled pathologies. These images were extracted using text annotation and query extraction methods and gathered from over 30,000 patients, who shall remain anonymous for security reasons. These images were then segregated into 15 classes, 14 different classes and normal classes. For clarity, peruse the image below to determine how each disease can be distinguished from the rest and the correlation between them.

3.1 Convolutional Neural Network Architecture

Convolutional Neural Networks[13] are feed-forward networks consisting of convolutional layers, pooling layers and fully connected layers with different activation functions suitable for individual layers as required. In this project, various pre-trained CNN models were used with different activation functions employed in order to maximise feature extraction efficiency and accuracy.

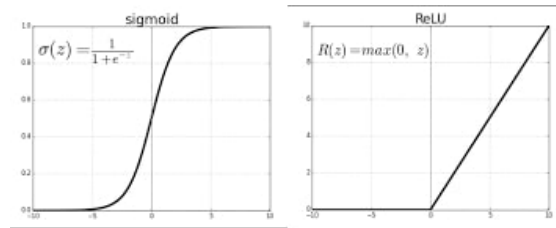


Figure 3: Graph plot of Activation Functions

Convolutional layers[14] are the building block of these CNN's. The operation is defined as the merging of two function in statistics. For ease of access, the images are first converted into matrices of equal size, and convolutional filter is applied to these images, which performs individual element-wise multiplication and stores the sum in a 3x3 feature vector. Numerous detector layers are operated with the input matrix which extract features to generate a layer of feature maps which forms an individual convolutional layer.

Activation functions: These are functions that are essential to determine whether a particular neuron in the neural network will activate or not, based on various mathematical functions. Different activation functions can be employed in order to maximize the accuracy and quality of neural networks. The activation functions used differ from each other, in order to maximize feature extraction and eliminate the vanishing gradient problem.

For the last dense layer, the binary cross-entropy[17] loss function will be used as opposed to the softmax function. This is because the probability of the 15 different classes should be independent of each other, which cannot be achieved while using the softmax function. Thus, the network will internally be able to create 15 different models(one for each class), and compute their probabilities independently. As a result, the sigmoid function was also employed to convert the multi-label classification into an n-binary classification problem.

These activation functions in combination are commonly used in developing CNN's as it is able to deal with the vanishing gradient problem and is useful for nonlinearity of layers.

Optimizers [9] are algorithms used to adjust the hyper-parameters of the neural network to minimise loss by changing the weights of the neuron and the learning rate, .The optimizer used in this paper is titled Adam[16] - short for Adaptive Moment Estimation, which uses both the function of RMS and momentum based gradient descent. Since our dataset involves a large number of parameters and data, Adam is most

useful since it is able to intuitively interpret loss functions and is able to tune the hyperparameter with minimal costs incurred.

Pooling layers are usually employed in CNNs to maximise pixel intensity of images. Global Average Pooling(GAP)[22] Layers are used in this paper to reduce the dimensionality and complexity of the image. This helps recognize only the most essential parts of the images.

Flattening and Fully Connected Layers are used to flatten the matrices, further reducing the dimensionality and producing a uniform feature vector which can be used for classification. This vector is then fed to the FC layer, which consists of multiple layers. All these layers are connected to each other extensively, and is used to extract features. This process is known as forward propagation, after which the loss function is calculated. The loss function used in this model is Cross-Entropy [10], since this is a classification problem. This function is then backpropagated using the optimizer Adam, and the weights are subsequently modified.

Additional Features also employed in this model are algorithms such as **Model Checkpoint**, **Reduce LR on Plateau and Early Stopping**. Model checkpoint helps maintain a record of the best accuracy scores and we can revert back to these hyperparameters if the model does not perform as expected. Reduce LR on Plateau is used to reduce the learning rate , in case the gradient descent ends up in a plateau and misses out on the global minima. Early stopping is used to avoid overfitting to the training data and is a form of regularization.

4 Model

4.1 MobileNet

MobileNet[7] is the first of four pre-trained models considered for this paper. MobileNet was developed in 2017 by TensorFlow and is regarded as its first mobile computer vision model. The base structure of MobileNet depends on an architecture that employs streamlined depth wise separable convolutions to construct neural networks that are extensive, but comparatively lightweight. It is able to exhibit lightweight neural network properties primarily due to the depth wise separable convolutions which reduce the number of parameters significantly. The architecture of MobileNet is set up as seen below.

Depth Wise Separable Convolutions [26] are extracted based on the principle of the Sobel Filter[27], which states that image matrices can be consequently separated into its vertical and horizontal components, and computed separately.

Type / Stride	Filter Shape	Input Size
Conv / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Conv dw / s1	$3 \times 3 \times 32$ dw	$112 \times 112 \times 32$
Conv / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$
Conv dw / s2	$3 \times 3 \times 64$ dw	$112 \times 112 \times 64$
Conv / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$
Conv dw / s1	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$
Conv dw / s2	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Conv dw / s1	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$
Conv dw / s2	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
5x Conv dw / s1	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 512$	$14 \times 14 \times 512$
Conv dw / s2	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$
Conv dw / s2	$3 \times 3 \times 1024$ dw	$7 \times 7 \times 1024$
Conv / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$
Avg Pool / s1	Pool 7×7	$7 \times 7 \times 1024$
FC / s1	1024×1000	$1 \times 1 \times 1024$
Softmax / s1	Classifier	$1 \times 1 \times 1000$

Figure 4: MobileNet Layers

This helps reduce the number of columns and filters required to process the images, and ultimately hastens the process. Here, two 3×3 matrices are used as filters which are then convolved with the image to generate their derivatives- two products for both horizontal and vertical components.

For example, consider the two filters

$$G_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

and

$$G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & 2 & 1 \end{bmatrix}$$

to be calibrated equivalent to the following values. Considering the original image to be A(matrix). When passed through a depth wise separable filter, the computations are as follows:

$$G_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} * A$$

$$G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & 2 & 1 \end{bmatrix} * A$$

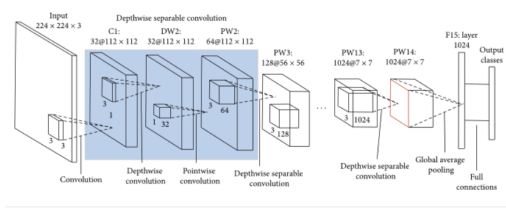


Figure 5: MobileNet Architecture

These filters can be further decomposed by passing a smoothing filter to produce the products of an averaging and differentiation kernel. As a result, The above equations can be further simplified as:

$$G_x = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} * \begin{pmatrix} 1 & 0 & 1 \end{pmatrix} * A$$

$$G_y = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} * \begin{pmatrix} 1 & 2 & 1 \end{pmatrix} * A$$

MobileNet employs 3x3 depthwise separable convolutions that results in about 8-9 times less computations than the standard convolutions. From these computations, we can see exactly why MobileNet is able to produce such lightweight networks with fast computations.

All layers of the MobileNet are followed by a Batchnorm and nonlinearity function in ReLU, with the exception of the final layers which utilizes a softmax layer for classification. Additional layers can be added based on image dimensions and degree of classification. The architecture of MobileNet is displayed in Fig 5.

4.2 ResNet

Residual Networks, or ResNets, were introduced in 2015 by He et al.[9] when the deep learning community faced the conundrum of whether implementing better learning networks was as simple as stacking more layers. Through research, it was formulated that as layers were added, training and error rates also increased as a result of vanishing/exploding gradient. As the number of layers increased, the accuracy flat-lined and subsequently started declining rapidly. This problem was addressed with the introduction of ResNets, with the addition of residual blocks helping in the elimination of the vanishing gradient problem.

Through the implementation of residual blocks, researchers have been able to overcome the gra-

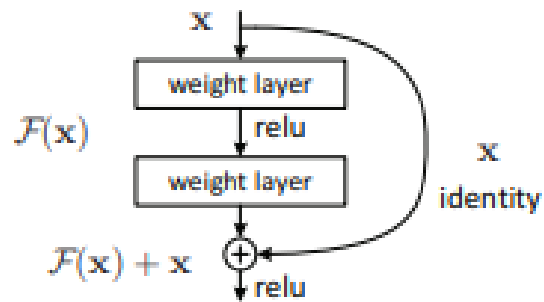


Figure 6: Structure of Residual Blocks

dient problem which has limited learning for a prolonged period of time. Residual blocks are based on a technique called **skip connections**, which essentially enables the network to skip some layers by connecting activations of initial layers to further layers, therefore bypassing some layers in between. This is especially beneficial because any layers that are found to affect the performance of the model can now be skipped, and ignored altogether by regularization.

The degradation problem faced by the extensive number of layers is addressed by designing a deep-learning residual framework, where the layers are explicitly fitted to a residual mapping. Formulating the activations of these layers as $H(x)$, it is mapped to another nonlinear mapping of $F(x) = H(x) - x$. As a result, the initial mapping of functions is now recast into $F(x) + x$. This is now simpler to optimize than the unreferenced, extensive mapping present in conventional networks. This formulation can be constructed by feedforward neural networks with **shortcut connections**, which enable the network to skip one or two layers. These connections perform the primary function of identity mapping, and the results of their computations are concatenated to the output of the stacked layers, as shown in Fig. 6.

There are a wide number of variants to the ResNet architecture, all titled differently. For this paper, ResNetX model is going to be implemented which consists of the following architecture(Fig. 7).

4.3 DenseNet

DenseNets[8] are another class of networks that were established to alleviate the vanishing/ exploding gradient problem that arose as a result of the extensive number of layers. While other models like ResNet and Highway connections[28] overcame this problem by implementing skip and shortcut connections, DenseNet

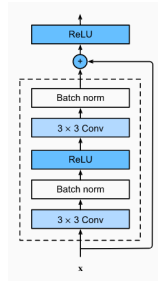


Figure 7: ResNet Architecture

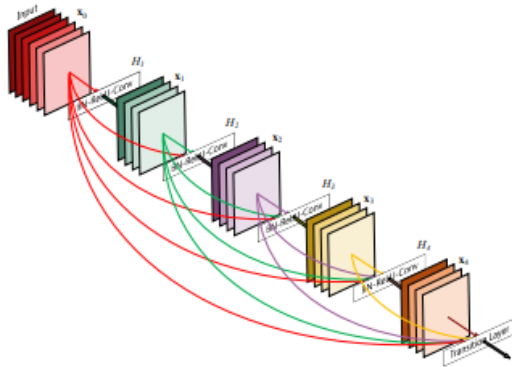


Figure 8: DenseNet Architecture

resolved this problem by simplifying the connection pattern between the convolutional layers and modifying the system architecture. In traditional CNNs, each layer receives the output of the previous layers in a feed-forward fashion, implying that for 'N' layers, there are 'N' direct connections between concurrent layers. DenseNet modifies these layer connections by simply connecting each layer directly with each other layer, hence the name - **Densely Connected Networks**. Therefore, there are $N(N+1)/2$ direct connections for layers in the DenseNet. This configuration ensures maximum information and gradient flow, by maximizing the applications of feature reuse.

With this configuration, the output feature maps in the DenseNet are thus not summed together, but instead concatenated and considered as inputs by the subsequent layers. As a result, DenseNet is able to function with fewer parameters than the orthodox CNN, paving the way for feature reuse, where the features which contribute the most to image classification are maximised and redundant feature maps are ignored. Consider the output of the layer 'l' which is a concatenation of the preceding outputs x_0, x_1 and so on, where x_2 is the output of layer n after composite functions have been performed on the image x_0 . Thus, x_l is formulated as follows:

$$X_l = H_l([X_0, X_1, \dots, X_{l-1}])$$

where $[X_0, \dots, x_{l-1}]$ is the result of the concatenation of the feature maps computed in the previous layers and H_l refers to the composite functions of operations like BatchNorm, ReLU and Pooling. These multiple computations of H_l can be densely packed into one single tensor for ease of implementation, hence justifying the title DenseNet.

5 Pipeline

The project pipeline follows the conventional Deep Learning approach, involving components such as Data Pre-processing, Model Training and Model Evaluation. The number of epochs determined for this model was fixed at 50 after training and testing, taking into account the computational resources associated with it and the size of the dataset. Additional modules such as EarlyStopping, ReduceLronPlateau, and ModelCheckpoint were also utilised to ensure that training is halted if there is no significant progress or if accuracy starts to decline after a set number of epochs. The best performing parameters are saved as best weights for reference and future usage. The system architecture is illustrated as follows:

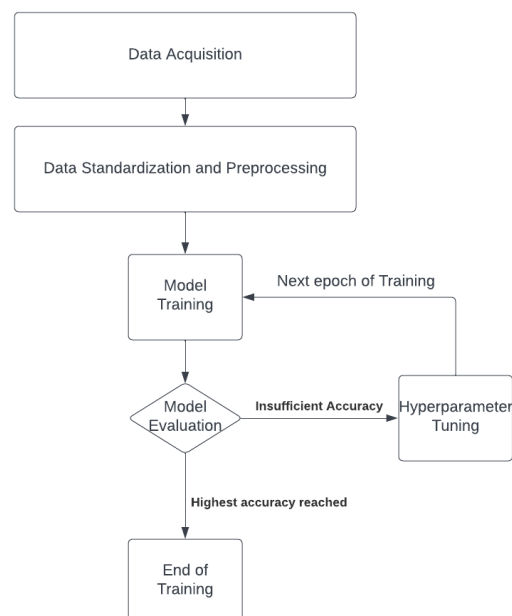


Figure 9: System Architecture and Design

5.1 Dataset

Chest Radiographs dataset titled 'Chest X-ray16' of 41.86 GB has been downloaded from

the Kaggle forum, released by the National Institute of Health, USA. This dataset consists of a total of 112,120 images consisting of 15 different classes. Each of these categories have been annotated by labels with a CSV file linking them.

5.2 Training

In the beginning, the dataset was examined for low quality images and possible outliers. Statistical analysis was initiated to determine if the data was skewed in one direction. Results showed that the data was skewed, hence normalization and regularization needed to be done. Data Augmentation was performed using the image process generator module and the dataset size was increased by altering the angle of viewing and so on. The radiographs were then fed to the different models in batches of 32 with image dimensions set as 224. The training was performed on the NVIDIA GeForce RTX 2070 with Max-Q configuration. Along with CUDA acceleration, the model processed 100 training images per batch, and 10 test images. The learning rate was initially set to 0.3, which was then modified accordingly as the epochs progressed.

6 Experimental Results

To evaluate the performance of the different models, accuracy, validation and testing loss, supplemented with sensitivity and specificity values which was then utilised to construct the AUC-ROC curve were the metrics considered. Accuracy is measured during each stage of the epoch where a batch of images is used to test how accurately the model is able to classify the images into their respective classes. Loss functions are calculated using back propagation and is an indicator of how far apart our predictions are from the actual target class.

Sensitivity is the metric that is responsible for displaying the proportion of positive classes that got correctly classified. It is also called true positive rate. Higher the sensitivity score, the better the model is at accurately classifying images. Specificity, on the other hand, is the metric that signifies the proportion of negative classes that got correctly classified. It is also known as true negative rate. Both these metrics are plotted in a graph to construct the Receiver Operating Characteristics (ROC) curve, where Area Under the Curve(AUC) is calculated at different thresholds. These metrics are calculated using the following formulae:

$$Sensitivity = TP/TP + FN \quad (3)$$

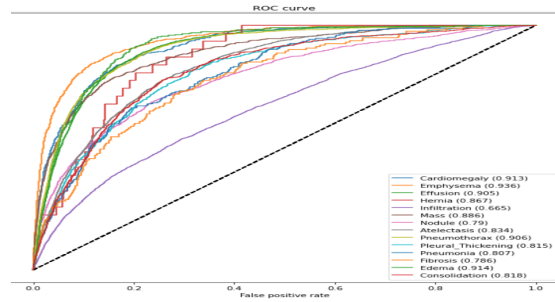


Figure 10: ROC Curve for DenseNet Configuration

$$Specificity = TN/TN + FP \quad (4)$$

$$FNR = FN/TP + FN \quad (5)$$

$$FPR = FP/TN + FP = 1 - Specificity \quad (6)$$

False Negative Rate(FNR) and False Positive Rate(FPR) are additional metrics which tells us the proportion of images that been incorrectly classified(both positive and negative classes).

The ROC curve is an evaluation metric designed for classification problems, which is essentially a probability curve that plots TPR vs FPR values respectively at different classification thresholds, with FPR on the X-axis and TPR on the Y-axis. This curve can be summarized to produce the AUC score, which is an indicator of the ability of the classifier to distinguish between classes. The ROC curve for this model is depicted in fig 10.

The different AUC values for each disease class and models can also be visualised in the form of a table, seen in table 1.

Disease Class	DenseNet AUC	ResNet AUC	MobileNet AUC
Cardiomegaly	0.913	0.53	0.825
Emphysema	0.936	0.54	0.856
Effusion	0.905	0.65	0.84
Infiltration	0.665	0.59	0.625
Hernia	0.867	0.59	0.862
Mass	0.886	0.52	0.704
Nodule	0.79	0.56	0.605
Atelectasis	0.834	0.60	0.72
Pneumothorax	0.906	0.52	0.839
Pleural Thickening	0.815	0.58	0.742
Fibrosis	0.786	0.61	0.688
Edema	0.914	0.74	0.842
Consolidation	0.818	0.64	0.749

Table 1: Disease Classes and their AUC scores

It can be inferred from table 1 that the various models performed well, with the AUC scores for each disease class being greater than 0.5 and close to 1. As a result, our models are able to distinguish between the different classes accurately. The DenseNet configuration achieved the highest AUC scores than the other models,

Model Design	Accuracy	Val Loss
CNN+VGG+STN[6]	0.677	0.699
VDSNet [23]	0.733	0.589
DenseNet121	0.887	0.376

Table 2: Model Performance Comparison

and is also worth noting that the model achieved an accuracy of 88.7 percent, higher than previously existing systems. Validation loss was also minimized as the epochs progressed, which is an indicator of a well-trained network.

7 Conclusion

This paper proposes different CNN models for accurately identifying normal and affected cases present in chest radiograph images. For optimal accuracy, the different CNN models was supplemented with algorithms such as Early Stopping, ReduceLR on Plateau, and Model Checkpoint to ensure that the computational costs are not exorbitant in the trade-off for high accuracy. As a result, the different CNN models were able to achieve near optimal scores on the basis of accuracy and AUC scores. Thus, it can be concluded that this CNN model can be used by medical practitioners and doctors as a diagnostic tool for early detection of pathologies and other aberrations. The CNN models was successfully implemented by utilizing various optimization algorithms like changing learning rates, adjusting the number of epochs and batch sizes and addition of fully connected layers to maximise back propagation. As a result, this model can be installed by medical practitioners to quickly process chest radiographs in order to detect lung diseases as early as possible in kids as well as adults. This ultimately helps increase the efficacy of the healthcare system in their time of diagnosis and treatment.

To verify the performance of our model, we cross-referenced the evaluation metrics of our neural network with other attempts to do the same; our algorithm was able to perform better than these algorithms, as seen in **Table 2**.

Future Enhancement

In the future, with more additions to optimise neural networks and advancements of computational resources, diagnostic models can be implemented which outperform every previous model and achieve near perfect accuracy. It can also be expected that neural network models can be coupled with ensemble, bagging, boosting and other revolutionary concepts such as genetic program-

ming in order to train and deploy high quality diagnostic models. To deal with vast sizes of data and images, the training process can be transferred to a cloud computing system, where multiple GPU's can be utilised to hasten the training process. A multidisciplinary effort where medical practitioners are able to configure these models and aid in feature extraction will also see a monumental increase in model accuracy and usage. Medical professionals would also need training to understand the functioning of these diagnostic tools available at their perusal.

References

- [1] Jeffrey, Ahmad- "Disease detection with deep Convolutional architecture"
- [2] Wang, X., Peng, Y., Lu, L., Lu, Z., Bagheri, M., Summers, R. M. (2017). ChestX-ray8: Hospital-scale Chest X-ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseases. ArXiv.
- [3] Ramalho GLB, Rebouças Filho PP, Medeiros FNS de, Cortez PC. Lung disease detection using feature extraction and extreme learning machine. Rev Bras Eng Bioméd [Internet]. 2014Sep;30(Rév. Bras. Eng. Bioméd., 2014 30(3)).
- [4] Jooae Choe, Hye Jeon Hwang , Joon Beom Seo, Sang Min Lee, Jihye Yun, Min-Ju Kim. Content-based Image Retrieval by Using Deep Learning for Interstitial Lung Disease Diagnosis with Chest CT
- [5] Naman Gupta a, Deepak Gupta a, Ashish Khanna a, Pedro P. Rebouças Filho b, Victor Hugo C. de Albuquerque. Evolutionary algorithms for automatic lung disease detection
- [6] Siddhanth Tripathi, Sinchana Shetty, Somil Jain, Vanshika Sharma - Lung Disease Detection Using Deep Learning
- [7] Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H. (2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. ArXiv.
- [8] Huang, G., Liu, Z., Weinberger, K. Q. (2016). Densely Connected Convolutional Networks.
- [9] He, K., Zhang, X., Ren, S., Sun, J. (2015). Deep Residual Learning for Image Recognition.
- [10] Ilyas, Maksim, Tamerlan - "Deep neural net-

- work ensemble for pneumonia localization from a large-scale chest x-ray database”
- [11] Jeremy Irvin, Pranav Rajpurkar -Deep learning for chest radiograph diagnosis
- [12] Karen Simonyan, Andrew Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition”, International Conference on Learning Representations, 2015.
- [13] Laith Alzubaidi, Jinglan Zhang, Amjad J. Humaidi-” Review of deep learning: concepts, CNN architectures, challenges, applications, future directions”
- [14] Saad Albawi; Tareq Abed Mohammed; Saad Al-Zawi - “Understanding of a convolutional neural network”
- [15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun - “Deep Residual Learning for Image Recognition
- [16] Diederik P. Kingma, Jimmy Ba - “Adam: A Method for Stochastic Optimization”
- [17] Zhilu Zhang, Mert R. Sabuncu - “Generalized Cross Entropy Loss for Training Deep Neural Networks with Noisy Labels”
- [18] Mir Mohammad Azad, Apoorva Ganapathy, Siddhartha Vadlamudi, Harish Paruchuri - "Medical Diagnosis using Deep Learning Techniques: A Research Survey"
- [19] Alhassan Mabrouk, Rebeca P. Díaz Redondo, Abdelghani Dahou – “Disease Detection on Chest X-ray Images Using Ensemble of Deep Convolutional Neural Networks”
- [20] C. Szegedy, W. Liu, Y. Jia et al., “Going deeper with convolutions,” in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1–9, Boston, America, 2015.
- [21] R. H. Abiyev and M. K. Ma’aitah, “Deep convolutional neural networks for chest diseases detection,” Journal of healthcare engineering, vol. 2018, 12 pages, 2018.
- [22] Arna Ghosh, Biswarup Bhattacharya, Somnath Basu Roy Chowdhury. AdGAP - Advanced Global Average Pooling
- [23] Bharati, S., Podder, P., Mondal, M. R. H. (2020). Hybrid deep learning for detecting lung diseases from X-ray images. Informatics in Medicine Unlocked, 20, 100391. <https://doi.org/10.1016/j.imu.2020.100391>
- [24] Bradley, A. P. (1997). The use of the area under the ROC curve in the evaluation of machine learning algorithms. Pattern Recognition, 30(7), 1145-1159.
- [25] Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., Xiong, H., He, Q. (2019). A Comprehensive Survey on Transfer Learning.
- [26] Chollet, F. (2016). Xception: Deep Learning with Depthwise Separable Convolutions.
- [27] N. Kanopoulos, N. Vasanthavada and R. L. Baker, "Design of an image edge detection filter using the Sobel operator," in IEEE Journal of Solid-State Circuits
- [28] Srivastava, R. K., Greff, K., Schmidhuber, J. (2015). Highway Networks. ArXiv.