Diagnosing Pneumonia from Chest X-Rays Using Neural Networks

Tushar Dalvi x18134301 Shantanu Deshpande x18125514 Yash Iyangar x18124739 Ashish Soni x18136664

***Abstract*—Disease diagnosis with radiology is a common practice in the medical domain but requires doctors to correctly interpret the results from the images. Over the years due to the increase in the number of patients and the low availability of doctors, there was a need for a new method to identify and detect disease in a patient. In recent years machine learning techniques have been very effective in image-based disease diagnosis. In this paper, a method has been proposed to diagnose Pneumonia using transfer learning with VGG19 model. For pre-processing online image, augmentation has been used to reduce the class imbalance in the training data, With this methodology. For pre-training the VGG19 model, Imagenet weights have been used and three layers have been stripped off the model and 2 dense layers have been added with Relu as an activation function in one layer and softmax in the other layer. We were able to achieve a recall of 0.96 and a precision of 0.87 with such a setting.**

***Index Terms*—Pneumonia detection, CNN, Image Classification**

I. Introduction

Radiology is one of the many branches of medicine that basically focuses on making use of medical images for the detection, diagnosis and characterization of the disease. One of the main responsibilities of a Radiologist is to view a medical image and based on the image producing a written report of the observed findings [1]. Medical experts have made use of this technique for around several years in order to visualize as well as explore abnormalities and fractures in the body organs. In recent years, due to the advancement in the technology field, we have seen an up-gradation in the healthcare systems. This has helped the doctors in better diagnosis accuracy and reducing the fatality rate. The chest is the most important part of the body as it contains the respiration organs which are responsible for sustaining important life function of the body. The count of people being diagnosed with a chest disease globally is in millions. Some of the diseases that are diagnosed using chest x-ray images are lung cancer, heart diseases, Pneumonia, bronchitis, fractures etc.

Pneumonia is one such kind of chest disease wherein there is an inflammation or infection in the lungs and it is most commonly caused by a virus or a bacteria. Another cause due to which a person can acquire Pneumonia is through inhaling vomit or other foreign substances. If a person is diagnosed with this disease, the lungs air sacs get filled with mucus, pus, and few other liquids which do not allow the lungs to function properly. This causes an obstruction in the path of oxygen to reach the blood and the cells of the body in an effective manner. Based on the data provided by the World Health Organization (WHO), 2.4 million persons die every year due to Pneumonia. [2]

One of the most tedious tasks for the radiologists is the classification of X-ray abnormalities to detect and diagnose chest diseases. As a result of this, several algorithms had been proposed by researchers to perform this task accurately. In these couple of decades, computer-aided diagnosis (CAD) has been developed through various studies to interpret useful information and assist doctors in getting a meaningful insight about an X-ray. Several research works were carried out in past by using artificial intelligence methodologies. Based on the learnings from artificial intelligence methodologies, we can also develop an image processing system that will use the X-ray images and will detect the diseases in the primary stages with improved accuracy. Some of the neural network methodologies that have been effectively used by the researchers in replacing the traditional pattern recognition methods for the diagnosis of diseases are probabilistic neural network (PNN), multi-layer neural network (MLNN), generalized regression neural network (GRNN), learning vector quantization (LVQ), and radial basis function. These deep neural networks have showcased increased accuracy in terms of image classification and have thereby motivated the researchers to explore more artificial intelligent techniques.

The convolutional Neural Network is another frequently used deep learning architecture due to its power of extraction of various different level features from the images. [3] By going through the relevant research studies, we hereby propose a deep convolutional neural network (CNN) in order to improve the accuracy of the diagnosis of chest disease, particularly Pneumonia.

II. Research Question

Can the Recall be improved for Chest X-ray Pneumonia Detection using CNN based VGG19 model as compared to the state of the art technique?

III. Literature Review

Owing to the difficulty in manually classifying the X-ray images for a disease, various studies have evolved around the use of Computer-aided diagnostics for obtaining meaningful X-ray information and thereby assist physicians to gain a quantitative insight of the disease. In this section, we will review the pieces of literature that supports the current research work.

In [4], the author examined three methods, Backpropagation NN, Competitive NN, Convolutional NN. In BPNN, the errors at output layers are propagated back to the network layer whereas CpNN is a two-layer NN based on a supervised algorithm. CNN has the strongest network as it can include many hidden layers and performs convolutions and subsampling in order to obtain low to a high level of features from the input data. Also, it has three layers; convolutional layer, subsampling/pooling layer, full connection layer. Results showed that CNN took more time and number of learning iterations however achieved the highest recognition rate and was also able to get a better generalization power over BPNN and CpNN.

In the above study, the output in terms of precision, computation time was not as efficient due to no prior image pre-processing performed. In [5], histogram equalization is performed for image pre-processing and the images are divided as well as normalized, further, cross-validation is conducted on test data. The proposed model achieved an accuracy of 95.3% on test data.

Further to the outstanding performance of CNN in [4], another study [6] was conducted that evaluated performance of three CNN architectures, Sequential CNN, Residual CNN and Inception CNN. In the residual, loss of information is prevented by passing information from earlier network layers downstream thus solving the problem of representational bottlenecks. Six residual blocks have been used and performance measured based on following metrics, accuracy, precision, AUC, specificity, recall. Non-linearity is introduced by using a Rectified Linear Unit (ReLU) layer. The VGC16 model outperformed all the other models. In [7], like [5], data has been normalized by dividing all pixels by 255 and thus transformed them to floating points. CNN model has compared with MLP wherein each node except the input node is a neuron that uses non-linear activation function. The technique involved multiplying inputs by weight and then add them to bias and dropout technique has been used to avoid overfitting and reduce the training time. CNN outperformed MLP based on evaluation metrics like cross-validation and confusion matrix.

A different approach can be seen in [8] where the network layer utilizes the final layer as the convolutional implementation of a fully connected layer that allows a 40-fold speedup. For dealing with imbalanced label distributions, a two-phase training procedure has been proposed. Performance of three different optimizers has been compared in [9], Adam optimizer, momentum optimizer and stochastic gradient descent. Filters are applied to the input image and the matrices thus formed are known as feature maps. Filters of 5x5 size used to extract features. In order to prevent overfitting, half of the neurons are disabled using the dropout technique. Adams optimizer outperformed the other two optimizers. Utilization of ResNet-50 for embedding the features has been done in [10] and CNN model thus built is able to get overall AUC of 0.816.

A different and novel approach is visible in [11] where the author used Transfer Learning to train the model. The advantage of TL is that a model can be trained even with limited data. A dataset of 5232 images has been used and the model achieved accuracy of 92.8% with sensitivity and specificity of 93.2% and 90.1% respectively. In [12], the pre-processing steps involved increasing the images and rotating them by 40 degrees. By following a similar model building process as [11], the author obtained the accuracy of 95.31% and is better than [11].

Two different CNN techniques were used in [13] and [14], the GoogleNet and AlexNet. During the pre-processing stage, multiple rotations at 90, 180 and 270 degrees have been performed and able to get AUC of 0.99. A rather different approach than CNN can be seen in [15] where the author used a technique called Multiple-instance learning (MIL). The research overcomes some of the drawbacks like no proper estimation of positive instances by reducing the number of iterations. The slow configuration is avoided by avoiding the reclassification and the AUC achieved is 0.86%. In [16], the author tackled the issue of class imbalance using image augmentation methods like rotating and flipping the images. In the last dense layer, the sigmoid function is used to avoid the problem of overfitting. Like [9], Adam optimizer has been used to reduce losses and batch size is set to 64 whereas epoch set to 100. The recall obtained is 96.7% and is better than state of art techniques. A 121 layer convolutional neural network has been trained on a large dataset of 100,000 X-Ray images with 14 diseases in [17]. Weights initialized using pre-trained weights from ImageNet and the model is trained using Adam. An extension of the algorithm is undertaken to detect multiple diseases and it is found that this model outperforms the state of the art.

A deep neural network algorithm, Mask-RCNN is used in [18]. This method uses lung opacity as a feature for binary classification and Pneumonia detection. The final output obtained is an ensemble of two models. The base network is trained with COCO weights based on ResNet50 and ResNet101 models to extract features from the images. An improved result due to an ensemble approach is 0.21. In [19], for extracting representative and discriminative features from the Chest X-ray images for the effective classification into various body parts, CNN's were explored and the capability of CNN in terms of capturing the image structure from feature maps is portrayed and hand-engineered features were easily outperformed by CNNs.

The usefulness of transfer learning [11] is further stated in [20] where the authors used a pre-trained ChexNet model with 121-dense layers and applied it in 4 blocks. The output of pretrained model is transferred to the new model and Adam optimizer is used. The model was trained on multiple blocks and multiple layers and best validation accuracy obtained was 90.38% for the model with 6 layers. The use of demographic features like age, weight and height is used in addition to image features in [21] and the balanced dataset is used for training. Five different algorithms have been used viz. InceptionResNetV2, DenseNet121, InceptionV3, ResNet50 and VGG19. The batch size used was 16 and Stochastic Gradient Descent was chosen as optimizer function. VGG19 model outperformed other models with AUC 0.9714 for training and 0.9213 for testing data.

Like [20] and [11], transfer learning is used in [22], however, in this study due to the high resolution of CXR images, an extra convolution layer is added. Proposed system showed enhanced screening performance. A multi-CNN model is used in [23] which consists of three CNN components. The output of each CNN component is a probability which classifies the image as normal or abnormal. The outcome of all three components is calculated by using the association fusion rule. The fusion rule consists of eight cases which lead to the final classification of the image. The model has been trained using the association fusion rule for classification of the images. Their model has achieved an accuracy of 96%.

IV. Methodology

From above Literature Review, we can say that in the medical domain it is hard for a layman to identify a disease by just looking at the images of the body part. To identify disease from x-rays, MRI Scans well-trained doctors or properly trained computer-based system are required because a layman doesn't possess the knowledge of the disease like the doctor. Even though we are building a model to detect Pneumonia we would want our model to detect different diseases with an adequate amount of training examples and model weights. So we will be using the CRISP-DM methodology [24]. It stands for Cross Industry Standard Process for Data Mining. It has six steps out of which we will be using Business Understanding, Data Understanding, Data Preparation, Modeling and Evaluation. We won't be deploying our model for real-world use as it is a model developed for this project and further methodologies need to be tested.

1) Setup: In this study all the data is image formatted, so we need to process the images to identify the patterns and based on that patterns need to classify the different classes. To perform all these tasks, we need a proper hardware system which can handle image processing task without system failure. Also, the requirement of proper GPU is must to train the model within minimum time. Python Programming Language has been used for this research, but due to lack of hardware processing power, normal laptops or systems might be not able to handle the model building process smoothly. To overcome this issue, for this study we are using cloud service-based platform named Google Collaboratory provided by Google. It is a free Jupyter notebook environment that requires no prior setup and runs entirely on cloud [25]. In this environment 12 GB of ram and 50 GB of storage space is provided initially as freemium service, also Google provides Free Tesla K80 Graphical processor of about 12 GB for fast computing. To perform this study data is uploaded on the Google drive for maximum time availability. And all the task which are mentioned in the next section is performed on Google Colab notebook.

V. Data Understanding and Pre-processing

While training the neural network, we face a common problem that not enough data is available to train the model to maximize the capability of the neural network [26]. In the medical domain data sets having more class imbalance as compared to a dataset of other domains. The data which we have gathered for our study is collected from Kaggle and is already divided into training, validation and testing sets. So we don't have to create the partition for the data. As mentioned above the data set is highly imbalanced, training data contains a total of 5216 images out of which 1341 are images of normal lungs and 3875 are images of Pneumonia infected lungs, which means for every image of normal lungs we have 3 images of Pneumonia infected lungs.

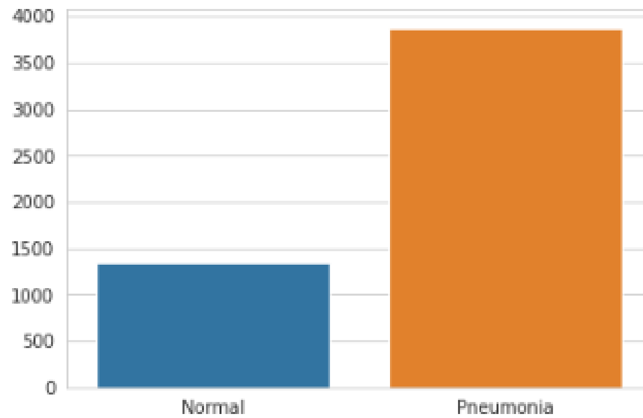

Fig. 1. Normal Vs Pneumonia Cases.

In such cases, the evaluation cannot rely only on metrics like Accuracy. There are many techniques available to reduce the recurring problem of class imbalance in the data which includes random undersampling, random oversampling and Synthetic Minority Over-sampling Technique(SMOTE). But there is a flaw in using these methods as there is a high probability of information loss as we have to detect Pneumonia and ignoring any Pneumonia case from the dataset will lead to loss of information for training purpose. Hence in data pre-processing Data augmentation is a technique used to deal with the class imbalance in the data without the need to collect new data. In this method the images of the class with fewer numbers in the data are duplicated by flipping, rotating, padding the available images of that class. There are two types of augmentation techniques one is offline and another is online, offline create a new dataset in the mentioned directory and it is time-consuming. But in Online Augmentation with the help of Keras, we can create a batch of images and that image gets duplicated and used as input, also online augmentation doesnt need that much space and it consumes less computational power. For code reproducibility we have set the initial seed to 100 with the function random.seet(100). In augmentation, we have performed rescaling, rotation, width shift, height shift, and horizontal flip using the ImageDataGenerator function from the Keras package [27]. In Rescaling, image is divided by 255 and converted into floating value. Also, images are rotated by 20 degrees. Width and height is shifted by 0.2 fractions of the total width, and with the horizontal flip parameter set to true, the images are flipped horizontally. The images in the dataset had varying size in height and width, which was not proper to provide as an input to the model, hence all images were downsized to 128 x 128 pixel by using the flow_from_directory function and also the shuffle flag parameter was set to true, so while training phase model should pick the random image form the data. Providing all images at the same time will take a lot of computational power to analyze images hence we set the batch size to 16 images per batch and set the class mode as categorical. With the help of the above steps, we were able to reduce the class imbalance in the dataset. For testing and validation data set only rescaling was performed during augmentation and also batch size was set to 16 images.

VI. Modeling

As this study aims to diagnose Pneumonia from Chest X-Ray images, a supervised learning technique is suitable for further study. Previous studies like [16] and [6] used Convolutional Neural Network and got promising results. A study conducted by [21] is used as our base paper to continue our research. As mention earlier in section IV-1 we are using Python programming language to complete this study, the main reason to use python is, it has large amount of libraries like NumPy, Matplotlib, and TensorFlow used to carry out various task, and with cloud environment like Google Collaboratory, and Google drive helps to keep personal system available for another task. To build a model it requires suitable computational power, with reference to [21] study, we are using VGG19 model for further process. Here VGG stands for the Visual Geometry group from the Oxford University who developed the VGG16 model for an image recognition competition and won it. The number '16' and '19' stands for the number of layers in the CNN model. This CNN model was developed and trained on multiple images for image classification. It consists of 3x3 layers of convolution stacked onto each other with the activation function as Rectifier Linear Unit (ReLU), to reduce the volumetric size max-pooling layers are used and are followed by fully connected layers again with ReLU as their activation function. It can classify images from almost one thousand different categories and can take an input image with dimensions 224x224 pixels. The model's application in this project is explained in the next section.

1) Model Building: Building a model or selecting a model is one of the crucial tasks in data mining techniques. Choosing the right model not only enhance the result but also reduce the time required to train the model. In this study, we use transfer learning to build out CNN model, in transfer learning a pre-built model store the knowledge which is gained while solving one problem and use and apply this knowledge on another related program [28]. In our study, we use MobileNetV2 as a base model for transfer learning. MobileNets improve the performance of mobile models on multiple tasks. We have created a custom model with the help of VGG19. As mentioned above it is a renowned Convolutional Neural Network Architecture for object recognition task developed and trained by Oxford's renowned Visual Geometry Group [29]. In this study VGG19() function from the Keras package imports VGG19 model. Here we use a novel approach by excluding the top 3 layers of the model, add two dense layers and we provide the input images with the dimensions of 128 x 128 x 3. Also, pretraining is performed on the Imagenet weights by allocating Imagenet attribute to the weights. Lastly, we set False flag for trainable attribute so while training the model no new weight will be updated. After importing VGG19 model we call the GlobalAveragePooling2D() function which performs global average pooling and calculates the average map of features from the previous CNN layer to reduce the effect of overfitting. Two dense layers are added to the model with activation function as ReLU and softmax respectively. The reason behind using ReLU activation function is, it is non-linear and doesn't produce back propagation errors like the Sigmoid function. It also speeds the model building process for a large Neural Networks. Softmax activation function helps to get output in 0 to 1 range, therefore the output from softmax falls under the probability distribution function and helps in classifying multiple classes as it outputs the probability for each class between 0 and 1. Also, while using a dense layer 512 units have been provided for the output space [30]. The Dropout rate is set to 0.7 to reduce the effect of overfitting.

2) Model Training: In the previous section, we build a custom model by adding 2 dense layers in the imported VGG19 model. But the model is configured for better performance and to minimize the loss function with the use of the compile function. While configuring compile function we need to use optimize our model to reduce training loss. In this study, Adam optimizer has been used because it computes the individual learning rates for different parameter. Secondly, for the loss function, categorical_crossentropy is used. categorical_crossentropy is a loss function that is used for categorization of single label that means the single example from our dataset should only belong to a single class. Hence our model will not classify the both Normal and Pneumonia class for a single image and for the metrics we have used accuracy for evaluation of the model on the training data. The model training phase is the most time-consuming phase in the entire data mining process. This phase takes time to perform depending on the size of the dataset and the number of iteration. To configure this process fit_generator() function from Keras is used to fit the entire training data into the memory, by setting the Epochs to 20 and verbose to 1. The main task of the fit_generator function is to discard the previous data which was already used for training purpose and create a new one for the next process. This task repeats the same process with the number of epochs that have been provided. Model training takes a lot of computing cost and its a heavy process when it comes to performing image-based data classification. Hence we run this process on Google collab with 12 GB of Ram and with Tesla K80 graphical processor with 12 GB graphical memory [31]. Even with this hardware configuration, it took around 9 hrs to complete 20 Epochs. While performing this training we store the log of each epoch to evaluate the performance of the model which is explained in the next section.

VII. Evaluation

In the previous section, we created our model and trained it on the training dataset, during the training period, logs were recorded to check the model performance. In this section, we will check the model performance from those logs. Evaluation technique helps to compare the generated result and expected result. Initially, with the help of logs, we will evaluate the accuracy and loss while training the model.

As we can see in the figure 4 loss on the Validation data is varying in all the epochs and gradually increasing, at the very first epoch loss reduces from 0.5 to 0.4 approximately. Till it reaches the 19th epoch it increases to 0.8 with lots of variations. In the last epoch loss on the validation data falls to 0.64. The loss of the training data constantly decreases. This contrast in the loss plot obtained by running the model with 20 epochs on both training and validation data suggests that the model is not overfitting the training data.

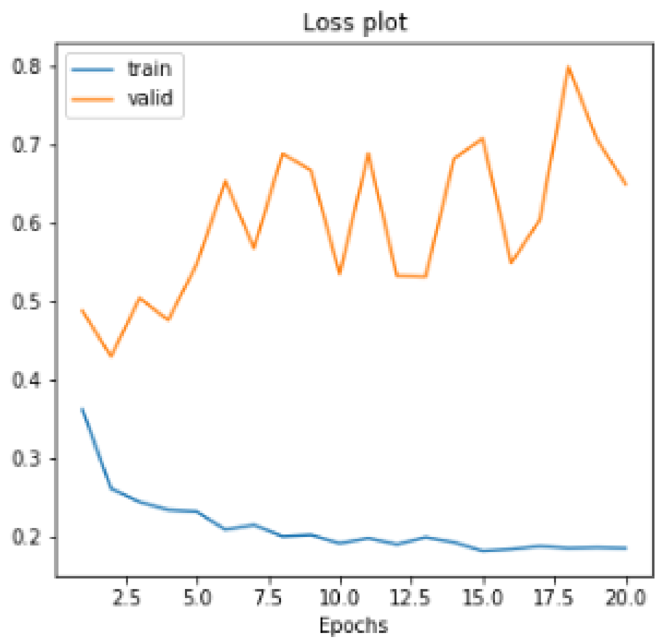

Fig. 2. Loss Plot

Accuracy plot tells us about how many images of both classes are being correctly predicted by the model. It increases for the training data gradually over the period of 20 epochs to approximately 0.92, but in case of validation dataset accuracy sharply increases from 0.81 to 0.875 and sharply decreases to .81 again and remains constant till 12 epochs and then again sharply varies till the last epoch. Even over here we can observe that both the training accuracy and validation accuracy do not flow in the same direction that is they do not increase constantly and together, this means that the model is not overfitting the training data's information.

The model's performance is evaluated over the test data using the evaluate_generator() function from the Keras package. This function checks accuracy on the testing dataset, in which maximum generator queue is set to 10 and set use_multiprocessing to true. As in result, we can see that loss on the test dataset is around 33% and accuracy is approximately 89%. But evaluate generator function initially

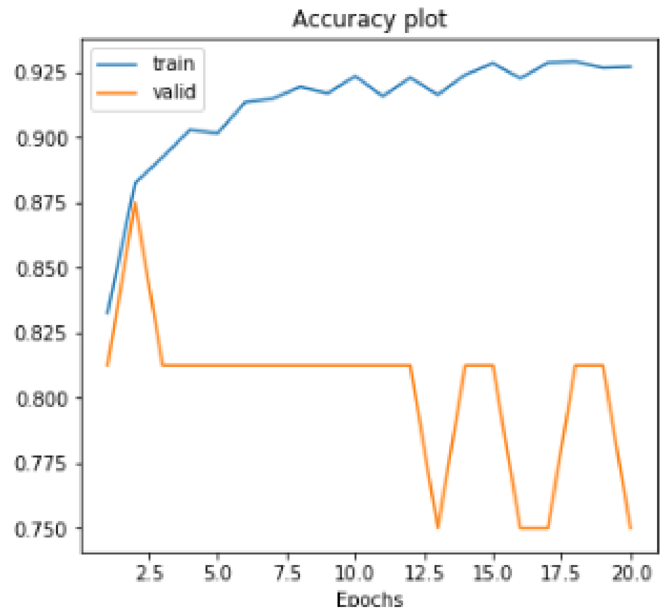

Fig. 3. Accuracy Plot

predict the output using training data and then evaluate the performance by comparing against the test dataset which we called as accuracy measure.

As mentioned in section V our dataset is from Health care domain and to calculate the proper prediction we need to consider recall and precision rather than accuracy. To calculate the recall and precision we need to provide the dataset which is not yet introduced to our model. For that, we need to perform prediction on the testing dataset. But before that, we need to identify the classes from test dataset to compare the result with our prediction. With the use of load_model() function from the Keras package, the saved model is loaded. And with the help of CV2 library, all images from testing dataset have been resized to 128 x 128 pixel to reduce the computational cost. Also with the help of CV2 library, each image has been labelled with their respective class which is 1 for Pneumonia infected lungs image and 0 for normal lungs image. This step helps to compare the actual image's class with predicted image's classes. To initiate the prediction process predict() function has been used. The model's performance on testing data is tested and a batch size of 16 and with the argmax function model returns the indices of the maximum values along the rows as it is set to 1. To plot the result confusion matrix is used with matplotlib library and precision and recall are calculated from the output which is discussed in the Result section.

VIII. Results

The performance of our classification model is described through a table known as the confusion matrix, as given below:

The basic terms to be interpreted from our confusion matrix are:

True Positives (TP) : The number cases where the model predicted yes and the person do have Pneumonia, i.e. 376.

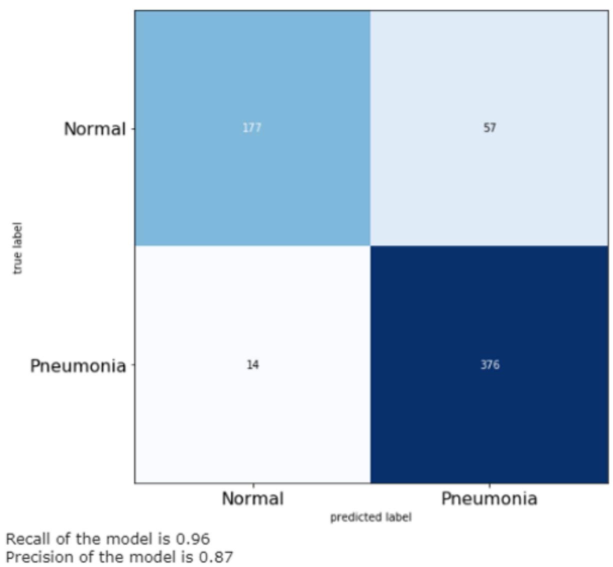

Fig. 4. Confusion Matrix

True Negatives (TN) : the number of cases where the model predicted no and the person does not have Pneumonia. i.e. 177. False positives (FP) : Model predicted yes, but actually they do not have Pneumonia. i.e. 57. False Negatives (FN) : Model predicted no, but actually they do have Pneumonia. i.e. 34.

How often our model is correct when it predicts yes, is given by precision and it is calculated as:

Precision = TP / (TP + FP) i.e. 0.87

Also, for our study, we will be focusing on Recall as a metric and expect it to be high. Recall would be a good measure of fit for our case as our goal is to correctly classify all the Pneumonia cases to save the patient's life. How many images the model has classified correctly out of all the positive classes is given by Recall and it should be as high as possible.

$$Recall = \frac{TP}{(TP + FN)}$$

As for our case, achieved Recall is 0.96. A high recall is expected but whenever there is a high recall, the precision reduces. This means an increase in the number of false positive. This means the model is wrongly classifying normal lung images as Pneumonia infected lung images. Thus there is a trade-off between recall and precision which needs to be maintained and depends on the domain and industry standards.

IX. Conclusion and Future Work

In this study, a methodology has been proposed to diagnose Pneumonia from chest x-ray images. The dataset selected had a high number of Pneumonia images and comparative less number of normal images. The imbalance ratio was 3 is to 1, so the image augmentation procedure was used to reduce the class imbalance. Due to time constraints and limited availability of resources other pre-processing methods like pixel brightness transformation and image segmentation could not be performed we suggest these techniques to be implemented in the future. The VGG19 model was used and three layers of the model were removed and two dense layers were added. Building a CNN model from scratch requires high computational capacity and as there was a shortage of time, we would suggest building a CNN model from scratch and fine-tuning the parameters for the future works. We were able to achieve a decent output with 0.96 recall, with a different approach, the result can be improved.

References

1 Kohli, M., Prevedello, L. M., Filice, R. W., and Geis, J. R., "Implementing machine learning in radiology practice and research," *American Journal of Roentgenology*, vol. 208, no. 4, pp. 754–760, 2017.

2 Er, O., Yumusak, N., and Temurtas, F., "Chest diseases diagnosis using artificial neural networks," *Expert Systems with Applications*, vol. 37, no. 12, pp. 7648–7655, 2010. [Online]. Available: http://dx.doi.org/10.1016/j.eswa.2010.04.078

3 Bengio, Y., Lamblin, P., Popovici, D., and Larochelle, H., "Greedy layer-wise training of deep networks," in *Advances in Neural Information Processing Systems 19*, Schölkopf, B., Platt, J. C., and Hoffman, T., Eds. MIT Press, 2007, pp. 153–160. [Online]. Available: http://papers.nips.cc/paper/3048-greedy-layer-wise-training-of-deep-networks.pdf

4 Tomè, D., Monti, F., Baroffio, L., Bondi, L., Tagliasacchi, M., and Tubaro, S., "Deep convolutional neural networks for pedestrian detection," vol. 2018, 2015. [Online]. Available: http://arxiv.org/abs/1510.03608{\%}0Ahttp://dx.doi.org/10.1016/j.image.2016.05.007

5 Saraiva, A., Ferreira, N., Lopes de Sousa, L., Costa, N., Sousa, J., Santos, D., Valente, A., and Soares, S., "Classication of Images of Childhood Pneumonia using Convolutional Neural Networks," pp. 112–119, 2019.

6 Rajaraman, S., Candemir, S., Kim, I., Thoma, G., and Antani, S., "Visualization and Interpretation of Convolutional Neural Network Predictions in Detecting Pneumonia in Pediatric Chest Radiographs," *Applied Sciences*, vol. 8, no. 10, p. 1715, 2018.

7 Hand, D. J., "Measuring classifier performance: A coherent alternative to the area under the ROC curve," *Machine Learning*, vol. 77, no. 1, pp. 103–123, 2009.

8 Havaei, M., Davy, A., Warde-Farley, D., Biard, A., Courville, A., Bengio, Y., Pal, C., Jodoin, P. M., and Larochelle, H., "Brain tumor segmentation with Deep Neural Networks," *Medical Image Analysis*, vol. 35, pp. 18–31, 2017. [Online]. Available: http://dx.doi.org/10.1016/j.media.2016.05.004

9 Hooda, R., Sofat, S., Kaur, S., Mittal, A., and Meriaudeau, F., "Deep-learning: A potential method for tuberculosis detection using chest radiography," *Proceedings of the 2017 IEEE International Conference on Signal and Image Processing Applications, ICSIPA 2017*, pp. 497–502, 2017.

10 Guan, Q. and Huang, Y., "Multi-label chest X-ray image classification via category-wise residual attention learning," *Pattern Recognition Letters*, no. xxxx, 2018. [Online]. Available: https://doi.org/10.1016/j.patrec.2018.10.027

11 Kermany, D. S., Goldbaum, M., Cai, W., Valentim, C. C., Liang, H., Baxter, S. L., McKeown, A., Yang, G., Wu, X., Yan, F., Dong, J., Prasadha, M. K., Pei, J., Ting, M., Zhu, J., Li, C., Hewett, S., Dong, J., Ziyar, I., Shi, A., Zhang, R., Zheng, L., Hou, R., Shi, W., Fu, X., Duan, Y., Huu, V. A., Wen, C., Zhang, E. D., Zhang, C. L., Li, O., Wang, X., Singer, M. A., Sun, X., Xu, J., Tafreshi, A., Lewis, M. A., Xia, H., and Zhang, K., "Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning," *Cell*, vol. 172, no. 5, pp. 1122–1131.e9, 2018. [Online]. Available: https://doi.org/10.1016/j.cell.2018.02.010

12 Stephen, O., Sain, M., Maduh, U. J., and Jeong, D. U., "An Efficient Deep Learning Approach to Pneumonia Classification in Healthcare," *Journal of Healthcare Engineering*, vol. 2019, 2019.

13 Lakhani, P. and Sundaram, B., "Radiol.2017162326," vol. 000, no. 0, 2017.

14 Gu, X., Pan, L., Liang, H., and Yang, R., "Classification of Bacterial and Viral Childhood Pneumonia Using Deep Learning in Chest Radiography," pp. 88–93, 2018.
15 Melendez, J., Van Ginneken, B., Maduskar, P., Philipsen, R. H., Reither, K., Breuninger, M., Adetifa, I. M., Maane, R., Ayles, H., and Sánchez, C. I., "A novel multiple-instance learning-based approach to computer-aided detection of tuberculosis on chest X-rays," *IEEE Transactions on Medical Imaging*, vol. 34, no. 1, pp. 179–192, 2015.
16 Liang, G. and Zheng, L., "A transfer learning method with deep residual network for pediatric pneumonia diagnosis," *Computer Methods and Programs in Biomedicine*, no. xxxx, 2019. [Online]. Available: https://doi.org/10.1016/j.cmpb.2019.06.023
17 Rajpurkar, P., Irvin, J., Zhu, K., Yang, B., Mehta, H., Duan, T., Ding, D., Bagul, A., Langlotz, C., Shpanskaya, K., Lungren, M. P., and Ng, A. Y., "CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning," pp. 3–9, 2017. [Online]. Available: http://arxiv.org/abs/1711.05225
18 Jaiswal, A. K., Tiwari, P., Kumar, S., Gupta, D., Khanna, A., and Rodrigues, J. J., "Identifying pneumonia in chest X-rays: A deep learning approach," *Measurement: Journal of the International Measurement Confederation*, vol. 145, pp. 511–518, 2019. [Online]. Available: https://doi.org/10.1016/j.measurement.2019.05.076
19 Srinivas, M., Roy, D., and Mohan, C. K., "Discriminative feature extraction from X-ray images using deep convolutional neural networks," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, vol. 2016-May, pp. 917–921, 2016.
20 Pardamean, B., Cenggoro, T. W., Rahutomo, R., Budiarto, A., and Karuppiah, E. K., "Transfer learning from chest x-ray pretrained convolutional neural network for learning mammogram data," *Procedia Computer Science*, vol. 135, no. Breast cancer: basic and clinical research 9 2015, pp. 400–407, 2018. [Online]. Available: https://app.dimensions.ai/details/publication/pub.1106388589andhttps://doi.org/10.1016/j.procs.2018.08.190
21 Heo, S. J., Kim, Y., Yun, S., Lim, S. S., Kim, J., Nam, C. M., Park, E. C., Jung, I., and Yoon, J. H., "Deep learning algorithms with demographic information help to detect tuberculosis in chest radiographs in annual workers' health examination data," *International Journal of Environmental Research and Public Health*, vol. 16, no. 2, 2019.
22 Hwang, S., Kim, H.-E., Jeong, J., and Kim, H.-J., "A novel approach for tuberculosis screening based on deep convolutional neural networks," 03 2016, p. 97852W.
23 Kieu, P. N., Tran, H. S., Le, T. H., Le, T., and Nguyen, T. T. T., "Applying multi-cnns model for detecting abnormal problem on chest x-ray images," *2018 10th International Conference on Knowledge and Systems Engineering (KSE)*, pp. 300–305, 2018.
24 Wirth, R. and Hipp, J., "CRISP-DM : Towards a Standard Process Model for Data Mining," *Proceedings of the Fourth International Conference on the Practical Application of Knowledge Discovery and Data Mining*, no. 24959, pp. 29–39, 1995.
25 Carneiro, T., Da Nobrega, R. V. M., Nepomuceno, T., Bian, G. B., De Albuquerque, V. H. C., and Filho, P. P. R., "Performance Analysis of Google Colaboratory as a Tool for Accelerating Deep Learning Applications," *IEEE Access*, vol. 6, pp. 61 677–61 685, 2018.
26 Lemley, J., Bazrafkan, S., and Corcoran, P., "Smart Augmentation Learning an Optimal Data Augmentation Strategy," *IEEE Access*, vol. 5, pp. 5858–5869, 2017.
27 Fujita, K., Kobayashi, M., and Nagao, T., "Data Augmentation using Evolutionary Image Processing," *2018 International Conference on Digital Image Computing: Techniques and Applications, DICTA 2018*, pp. 1–6, 2019.
28 Ling Shao, Fan Zhu, and Xuelong Li, "Transfer Learning for Visual Categorization: A Survey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 5, pp. 1019–1034, 2014.
29 Shaha, M. and Pawar, M., "Transfer Learning for Image Classification," *Proceedings of the 2nd International Conference on Electronics, Communication and Aerospace Technology, ICECA 2018*, no. Iceca, pp. 656–660, 2018.
30 Lau, M. M. and Lim, K. H., "Review of adaptive activation function in deep neural network," *2018 IEEE EMBS Conference on Biomedical Engineering and Sciences, IECBES 2018 - Proceedings*, pp. 686–690, 2019.
31 Garg, A., Gupta, D., Saxena, S., and Sahadev, P. P., "Validation of Random Dataset Using an Efficient CNN Model Trained on MNIST Handwritten Dataset," *2019 6th International Conference on Signal Processing and Integrated Networks, SPIN 2019*, pp. 602–606, 2019.