Classification of Chest Radiography Scans for COVID-19

Navya Agarwal
Department of Artificial Intelligence and Data Science
Indira Gandhi Delhi Technical University for Women
Delhi, India
navya.agarwal.in@gmail.com

Ananya Srivastava
Department of Artificial Intelligence and Data Science
Indira Gandhi Delhi Technical University for Women
Delhi, India
sriv.ananya02@gmail.com

Poonam Bansal
Department of Artificial Intelligence and Data Science
Indira Gandhi Delhi Technical University for Women
Delhi, India
poonambansal@igdtuw.ac.in

Kiran Malik
Department of Artificial Intelligence and Data Science
Indira Gandhi Delhi Technical University for Women
Delhi, India
kiranmalik@igdtuw.ac.in

***Abstract*— Humanity has suffered as a result of the COVID-19 pandemic for more than two years. Testing kits were not widely accessible during the pandemic, which caused alarm. Any technical development that enables a quicker and more accurate identification of COVID-19 infection can be very beneficial for the medical field. X-rays can be used to examine a patient's lungs since COVID-19 targets the epithelial cells that line the respiratory system. It is challenging to determine COVID-19 from other Viral Pneumonia cases, though. The purpose of this paper is to examine the effectiveness of deep learning models in the quick and precise detection of COVID-19 in chest X-ray scans.**

Keywords*— Deep learning, Image Classification, Computer Vision, COVID-19, CNN

I. Introduction

The COVID-19 pandemic has exposed significant challenges for the healthcare infrastructure and medical research communities, just like the two previous emergences of coronavirus disease in the past 18 years, "SARS (2002 and 2003)" and "Middle East respiratory syndrome (MERS) (2012 to the present)".

The diagnosis of the infection could not be done at the pace the infection was spreading due to the lack of testing kits. This created the need for a much faster method of diagnosis that could be carried out quickly for a large number of people. Chest X-Ray is one such option. However it comes with its own set of challenges, such as X-Ray scans of COVID-19 being similar to those of Viral Pneumonia and lack of availability of large enough data sets for training of complex deep learning models.

This paper first surveys the existing work done in the field and then talks about our own deep learning models that investigate the best methods to go about classifying COVID-19 chest X-ray scans.

Before implementing our models, we carry out an elementary data analysis, which reveals the significant class imbalance within the data set. We perform data augmentation to get rid of this imbalance in one of our models. Augmenting data instead of random oversampling is a better way to handle class imbalance as it prevents over-fitting.

Next we implement Convolutional Neural Network models which are best for working with images. We implement a CNN without data augmentation and another one with data augmentation. We add regularisation and dropout layers to our architecture to prevent model over-fitting, and to generalise our model.

The last model that we implement is a hybrid of the VGG-16 pre-trained model. We defined top layers for the hybrid that included several pooling and dropout layers. In addition, we added our own hidden and output layers. This hybrid model best demonstrates transfer learning, an imperative domain that helps one build on already existing machine learning algorithms and architectures.

In the end, the training accuracy, training loss and testing accuracy of the various models is compared and conclusions are formed about the various techniques used.

II. Background

We provide a background from some research publications that aim to accurately classify chest X-Ray scans using various Deep Learning approaches in this section. Some of the commonly discussed approaches utilise CNNs with various modifications, transfer learning and anomaly detection.

[1] [2] Asmaa Abbas et al. validate a "deep convolutional neural network (CNN) called Decompose, Transfer, and Compose (DeTraC)" for the classification of COVID-19 chest X- ray images in their paper titled "Classification of COVID-19 in chest X-ray scans using DeTraC deep CNN". By exploring the class boundaries with a class decomposition technique, "DeTraC" can handle any irregularities in the image data set. "DeTraC" was successful in identifying COVID-19 chest X- ray images from cases with "mild acute respiratory syndrome" and cases with "severe acute respiratory syndrome" with a high accuracy of 93.1%.

[3] Sohaib Asif et al. propose the "DCNN-based model Inception V3 with transfer learning" in their paper titled "Classification of COVID-19 from Chest X-ray images using Deep CNNs" for the detection of patients with coronavirus using chest X-ray radiography. This model achieves a classification accuracy of greater than 98%. The outcomes reveal that transfer learning for COVID-19 detection is a technique that works well, performs consistently, and is simple to implement.

[4] Joy Iong-Zong Chen's research study has developed a classification approach for carrying out accurate detection of COVID-19 using histogram-oriented gradients feature extraction methodology. Based on several edge-based neural networks, the effectiveness of their proposed CNN

classification approach for medical imaging has been evaluated. The accuracy of tertiary classification with CNN will reduce when there are more classes in the training network. They reported that their proposed model could classify normal, COVID-19, and pneumonia X-ray images with 85% accuracy.

Self-supervised learning, an alternative to transfer learning, involves changing images that are unlabeled from target domain themselves into a supervised task. In some ways, the lower layers are pre-trained to invert the transformation. [5] Deepak Anand et al. demonstrate in their work that "self- supervised learning coupled with adversarial training" provides many more benefits over transfer learning and simple self-supervised learning. In particular, "adversarial training" produces a natural robustness to adversarial attacks as well as a minimization in the amount of supervised data needed for equivalent accuracy.

By using a DNN model and only "normal" images for training, [6] Takahiro Nakao et al. present an "unsupervised anomaly detection" method that can be tested on a large chest radiography dataset. As a DNN model, they employed the "auto-encoding generative adversarial network framework", which combines a variational autoencoder and a GAN. By training with only the normal images, their unsupervised anomaly identification algorithm could accurately identify various anomalies and diseases in chest radiographs.

The [7] "confidence-aware anomaly detection (CAAD) model, which consists of a shared feature extractor, an anomaly detection module, and a confidence prediction module", is proposed by Jianpeng Zhang et al. in their publication. They accept the input as an anomaly case if the confidence score estimated by the confidence prediction module is minimal enough or the anomaly score generated by the anomaly detection module is large enough (that is Viral Pneumonia). The main advantage of their method over "binary classification" is that they avoid explicitly modeling several classes of Pneumonia and consider all reported cases of Pneumonia as anomalies in order to support the one-class model.

A number of research papers also explore [10] [12] various modifications based off the U-Net pre-trained model for lung images segmentation. [11] Sifat Ahmed et al. trained their U- Net based model called HRNet on an extensive dataset and achieved significant results. [12] Adnan Saood Iyad Hatem conclude from their work that while SegNet is better at segmenting images as infected or non-infected, U-Net based networks achieve superior performance in multi-class classification.

III. Methodology

Three crucial steps make up the study's workflow (refer Fig. 3). The goal is to implement different models, which we further use to contrast how each of the various techniques perform.

A. Dataset Used

A dataset of [8] [9] "chest X-ray images for COVID-19 positive cases, along with normal and viral pneumonia images, has been created by a group of researchers from the University of Dhaka, Bangladesh and Qatar University, Doha, Qatar, as well as their collaborators from Pakistan and Malaysia." They worked with medical professionals to create this dataset which was then accessible on Kaggle.

It has 3616 positive COVID-19 scans, 10216 scans of the normal lungs, and 1345 scans for viral pneumonia. Refer Fig. 1 for examples of lung scan images in the dataset.

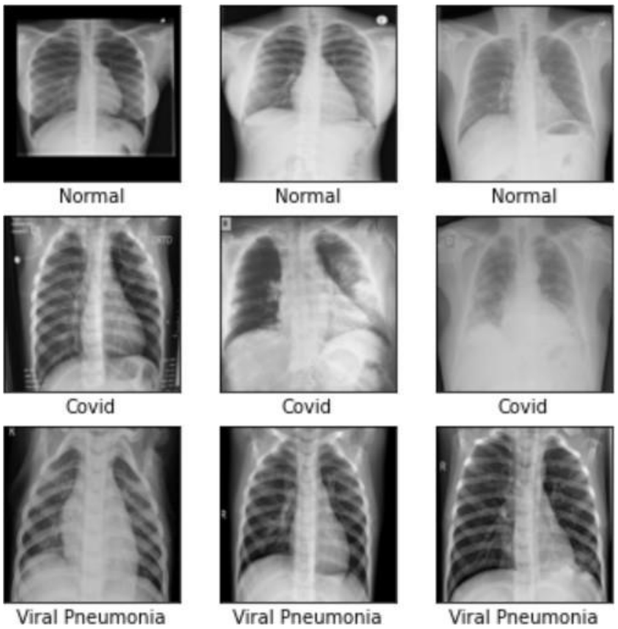

Fig. 1. Data Sample

B. Data Analysis

To uncover some critical insights, rudimentary data analysis was applied to the dataset. Every image is of data type uint8 and of dimensions 299x299. Even though these are black and white X-ray images, there are 3 colour channels - red, green and blue.

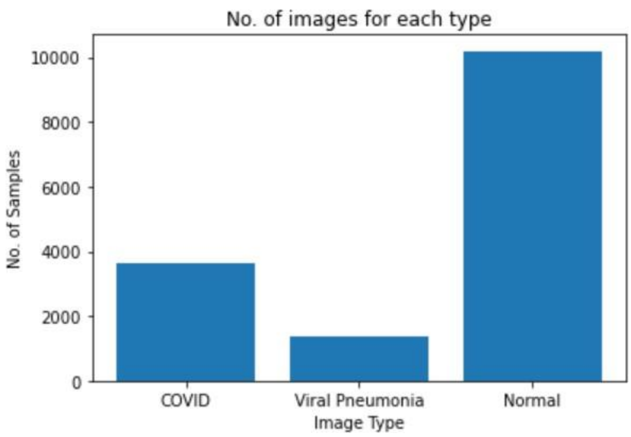

Fig. 2. Bar Graph depicting class imbalance in dataset

As observed in Fig. 2, there is an imbalance between the number of samples in the three classes of the dataset. A large number of neural network models used for multiclass classification have been built with the pre-requisite that the number of samples in each class is approximately equal. Thus, class imbalance can give us skewed results after prediction, especially during the classification of a minority class.

We propose three different models: a 13-layered Convolutional Neural Network architecture trained on the base dataset, a similar model trained on an augmented dataset and the last model which showcases transfer learning.

IV. Implementation

In this section, we discuss the detailed implementation for each of the three models trained on the chest X-Ray scan dataset.

A. *CNN Model without data augmentation*

Data loading and Pre-processing: We create a function loadImages that reads images from directories and adds them to a list and also creates a list of their target labels. It performs normalisation and resizes the images to dimensions of 100x100 for ease of processing.

The labels for the three classes are decided as follows: Normal = 0, COVID-19 = 1, Viral Pneumonia = 2. The images for the specific labels are stored in three separate lists for normal, COVID-19 and Viral Pneumonia labels using the loadImages function. These three lists and their target labels lists are stacked into two lists – data and targets.

Data Splitting: Using the train-test-split from Sklearn, the entire set of data is divided into training, validation, and testing data sets with proportions of 70%, 10%, and 20% respectively.

We build a CNN model using Sequential from sklearn with the specifications given in Table I. Regularizers are added to the convolutional layers, and dropout layers are also added in the model to prevent overfitting.

Regularizers: Technique that helps the model to generalise better so that it gives better accuracy on unseen/ test data.

[12] **Dropout layers:** By disregarding some randomly chosen neurons during the training phase – that is, by not taking these neurons into account during a specific forward or backward pass—dropout layer helps to prevent overfitting. This forces all neurons to improve their ability to generalise and prevents the network from depending too much on individual neurons.

The “adam” optimizer is used to build the model, while “Sparse Categorical Cross Entropy” is used as the loss function. Accuracy is the metric that has been selected for training.

ADAM: “Adaptive Moment Estimation” (ADAM) is suggested as the default optimizer for a majority of applications since its results are typically superior to most other optimization algorithms, it has a quicker computation time, and it requires less tuning parameters.

Sparse Categorical Cross Entropy: When there are two or more classes in our classification work, we use this. Additionally, it necessitates that our labels have the form of integers. Contrary to Categorical Cross Entropy, this loss function does not need one-hot encoding.

Then the model is trained with a batch size of 64 and the maximum number of epochs is set to 50. A call-back is also passed to the model to stop training the model early in case the validation loss does not decrease in 3 consecutive epochs.

For simple reloading, we save this model in HDF5 format. It includes details of the model’s compilation history and its weight values. It is a simpler alternative of SavedModel.

TABLE I. 13-Layered CNN Architecture

Layer (Type)	Output Shape	Param #
conv1 (Conv)	(None, 98, 98, 32)	896
maxpool1 (MaxPooling Layer)	(None, 49, 49, 32)	0
drop1 (Dropout Layer)	(None, 49, 49, 32)	0
conv2 (Conv)	(None, 47, 47, 32)	4624
maxpool2 (MaxPooling Layer)	(None, 23, 23, 16)	0
drop2 (Dropout Layer)	(None, 23, 23, 16)	0
conv3 (Conv)	(None, 21, 21, 16)	2320
maxpool3 (MaxPooling Layer)	(None, 10, 10, 16)	0
drop3 (Dropout Layer)	(None, 10, 10, 16)	0
flatten (Flatten Layer)	(None, 1600)	0
dense1 (Dense Layer)	(None, 512)	819712
dense2 (Dense Layer)	(None, 256)	131328
dense3 (Dense Layer)	(None, 3)	771

B. *CNN Model with data augmentation*

While performing Exploratory Data Analysis, we see that there is a significant class imbalance between the normal, COVID-19 and viral pneumonia classes. To eliminate this imbalance, we perform data augmentation using the ImageDataGenerator() from sklearn. Then we create an iterator object for it that is passed to the model during compilation.

The rest of the steps are performed similar to the previous model.

C. *VGG-16 Pre-Trained Model*

Keras Applications is a deep learning framework that supplies neural network models as well as pre-trained weights. These models are utilised in projects for extracting features, predicting, and fine-tuning. A convolutional neural network with 16 layers called VGG-16 is what we used here.

The utilised model has “already been trained using data from the ImageNet database containing more than a million photos. The pretrained network can categorise photos into 1000 different object categories, including several animals, a keyboard, a mouse, and a pencil.” This network has gained vast feature representations for a wide variety of visual data. The input for this model should be images with a resolution of 224x224.

The images were present in different directories on the basis of the class they belonged to, so we created a list containing the path of each image file in our dataset and another list containing the label of the image which we extracted from the directory name in which it was present.

Then we created a dataframe using Pandas with 2 columns: a ‘File Path’ column containing the paths of the images and a ‘Category’ column which contained the label of each image. Using SciKit’s “train-test-split” method, we divided the data into training, validation, and testing images, dividing the data into a ratio of 75%, 12.5%, and 12.5%. This technique randomly divides our dataset into several strata or categories based on the image labels.

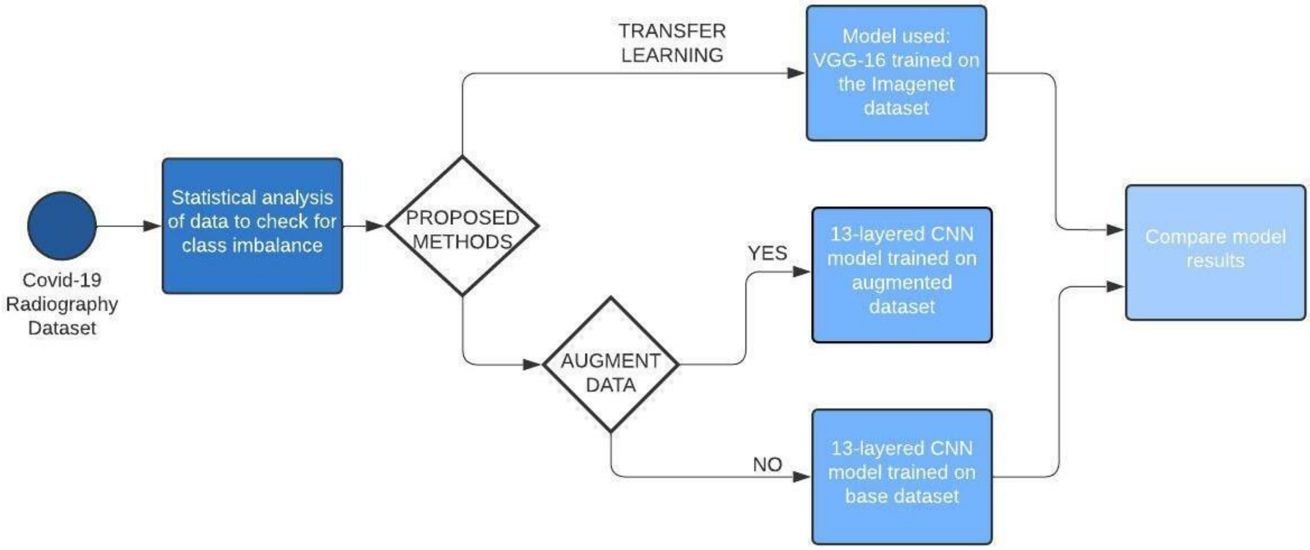

Fig. 3. Flowchart of Methodology

Image Pre-processing: With real-time data augmentation, the Image-Data-Generator method was utilised to generate batches of tensor image data (using a data frame).

By applying random (but believable) modifications, such as image rotation, to our training set, we can use a technique called data augmentation to make it more diverse. We also performed normalization techniques such as rescaling our images. Here are a few parameters that we changed:

1) class-mode = set to 'categorical', the y-col parameter contains the target labels/classes

2) x-col = column in dataframe that contains file paths of images

3) target-size = size to which all images are resized

We created a training data generator and a similar validation data generator. We instantiated a VGG16 Model (described above) where we left out the fully connected top layers because we wanted to connect the pre-trained model layers to those that we defined. The layers that we added to the base model are:

4) Pooling Layer: An average pooling layer.

5) Flatten: A layer that flattened the multi-dimensional images to 1D arrays.

6) Dense: A hidden layer with 128 neurons and the Rectified Linear Unit activation function.

$$\text{Relu}(x) = \max(0, x)$$

7) Dropout: For the prevention of overfitting, the Dropout layer randomly sets inputs after every 'f' intervals to 0.

The overall sum of all the inputs still remains equal to the previous sum because the non-zero inputs are scaled up by multiplying them with a factor of 1 / (1 - f).

8) Dense: An output layer containing 3 neurons which is using the Softmax activation function.

Then, using the Categorical Cross Entropy Loss (commonly known as the Softmax Loss), we assembled our model. It is a Cross-Entropy loss combined with a Softmax activation. If we employ this loss, we will train our model to produce a probable class for each image across the n classes. It serves as a tool for multiclass classification. We utilised Adam as our optimizer.

The accuracy of the model was selected as the performance metric since it calculates the adaptive learning rate for each parameter.

V. Result

The results obtained from the training and validation of the aforementioned models are discussed in this section.

CNN model without data augmentation attains a training data accuracy of 94.51% and validation data accuracy of 91.69% and we see that the model is well fitting (refer Fig. 4 and Fig. 5), that is, both under and over-fitting of the model are avoided.

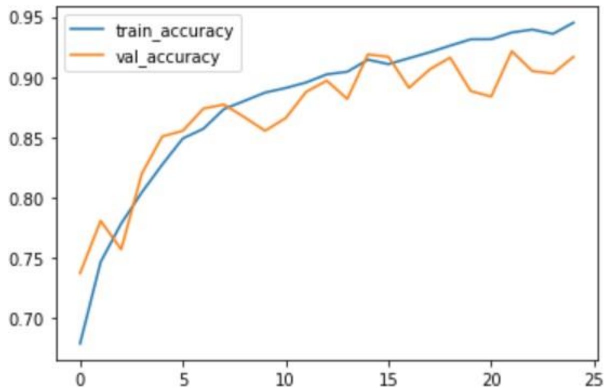

Fig. 4. Training & validation accuracy for CNN model without data augmentation

Then we predict the target values for the testing data. The accuracy for the test data points on the trained model is found to be 91.55%.

Measuring accuracy alone can be misleading for classification algorithms when we have an unbalanced number of observations in each class or for more than two classes in the data set. Thus, we plot the confusion matrix for both the CNN models to compare their performance reliably (refer Fig. 6 and Fig. 9).

Next, we plot the accuracy and loss values for both training and validation data for the CNN model with data augmentation (refer Fig. 7 and Fig. 8).

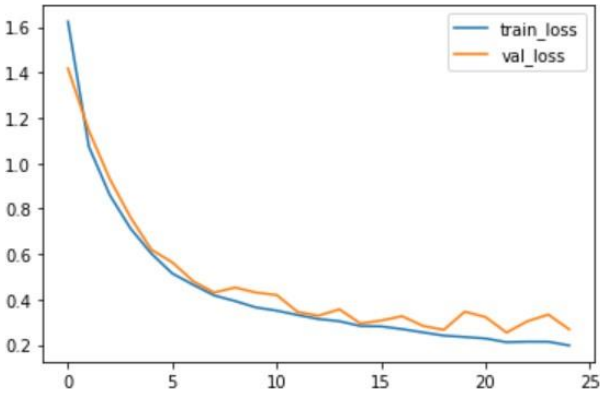

Fig. 5. Training & validation loss for CNN model without data augmentation

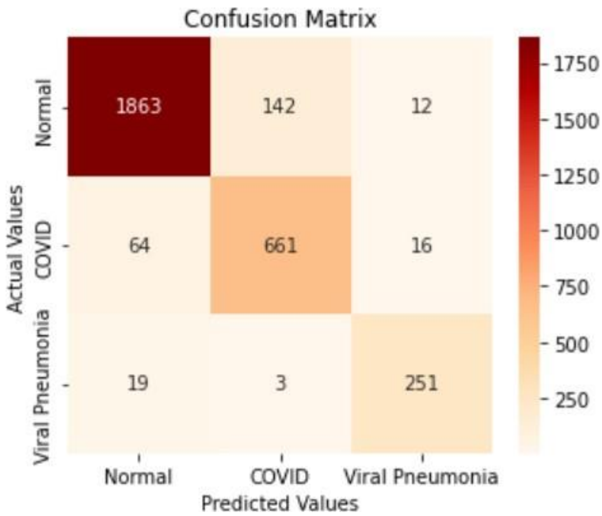

Fig. 6. Confusion matrix for CNN without data augmentation

The training and validation accuracies for this model are 93.15% and 91.03% respectively.

We see that the values for both CNN models are comparable. However, the CNN model with data augmentation achieves its target of reducing the mis-classification in the minority class, which in this case is the Viral Pneumonia class.

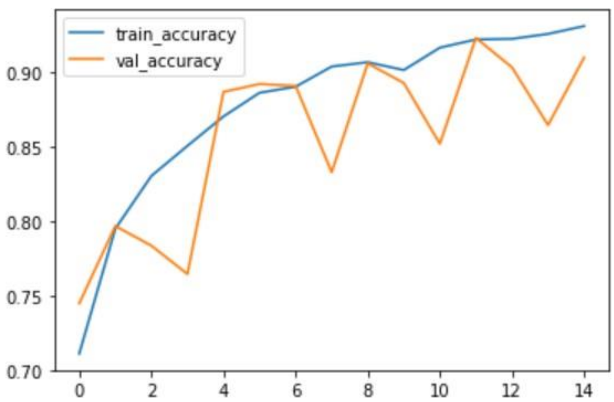

Fig. 7. Training & validation accuracy for CNN model with data augmentation

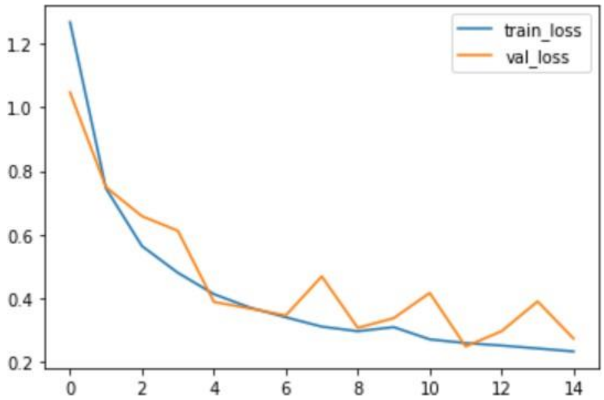

Fig. 8. Training & validation loss for CNN model with data augmentation

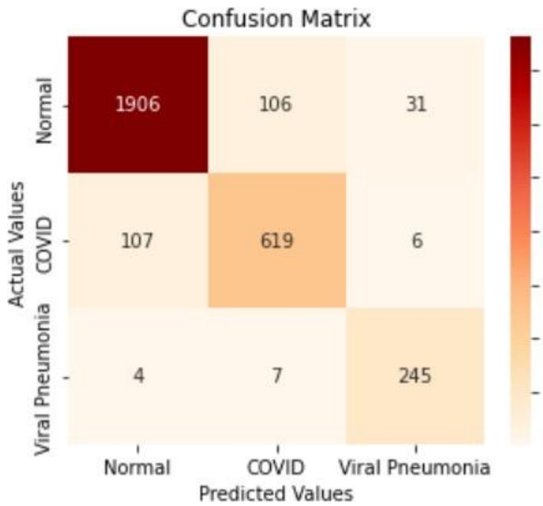

Fig. 9. Confusion matrix for CNN with data augmentation

For the pre-trained model using VGG-16, we attain the highest training data accuracy of 91.47% and the highest validation data accuracy of 93.98% after 10 epochs (refer Fig. 10 and Fig. 11).

The VGG-16 hybrid model is more complex than the CNN model that has been implemented because it is a 16-layered architecture as opposed to the CNN which has 13 layers. Even so, the accuracy of the hybrid is comparable to the simpler CNN because the model has been trained using the ImageNet database, while the latter was trained on the specific dataset that we used.

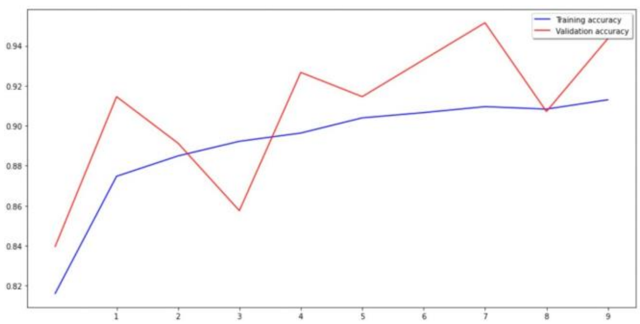

Fig. 10. Training & validation accuracy for VGG-16 hybrid model

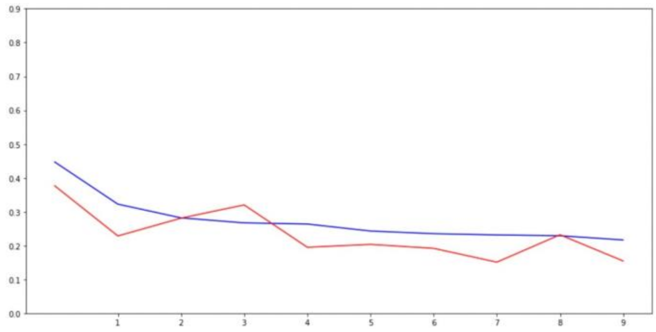

Fig. 11. Training & validation loss for VGG-16 hybrid model

```
True value is : Normal
Prediction is:
Normal
<matplotlib.image.AxesImage at 0x7f693eee7710>
```

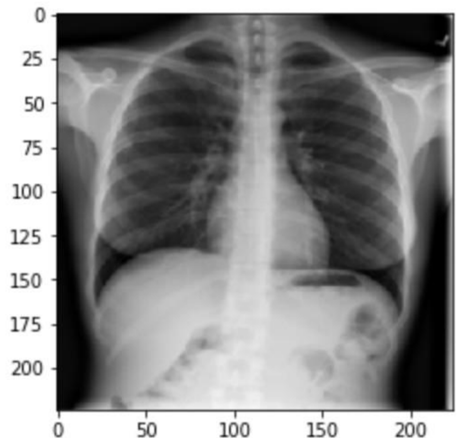

Fig. 12. Model Testing

The VGG-16 hybrid model is tested on a sample scan, and the truth value of the sample scan turns out to be the same as the predicted value (refer Fig. 12).

VI. Conclusion and Future Scope

After a thorough analysis of our implemented models and the literature survey, we can conclude that transfer learning combined with methods that handle class imbalance and prevent over-fitting provide a much higher accuracy in classifying COVID-19 chest X-Ray scans from those of normal and Viral Pneumonia X-Ray scans.

Given the enormous amount of computing power and time needed to create neural network models for these problems, transfer learning provides us with a very good place to start when working on problem statements related to computer vision.

Our models also investigated the use of image data augmentation to improve performance. We conclude that augmenting data instead of random over-sampling is a better way to handle class imbalance as it prevents over-fitting.

Google Research's initiative, AI for Medical Imaging, previously worked on CXR classification into normal versus abnormal lung scans for two diseases, tuberculosis and COVID-19, using a large dataset collected from Apollo Hospitals across India. This is a testament to the tech industry's growing research sector. Their model was used for binary classification while the models in our paper have been used for multi-class classification.

Both of our models exhibit the use of deep learning to predict respiratory diseases, which can be extrapolated for other diseases as well. While there are huge strides being taken in the field of Computer Vision for medical science, there is a lot of room for improvement. Collection and open source availability of more data sets in the field would encourage more work in this direction.

References

[1] Abbas, A., Abdelsamea, M.M. Gaber, M.M., Classification of COVID-19 in chest X-ray images using DeTraC deep convolutional neural network. Appl Intell 51, 854–864(2021), https://doi.org/10.1007/s10489-020-01829-7

[2] A. Abbas, M. M. Abdelsamea and M. M. Gaber, 'DeTrac: Transfer Learning of Class Decomposed Medical Images in Convolutional Neu- ral Networks', in IEEE Access, vol. 8, pp. 74901-74913, 2020, doi: 10.1109/ACCESS.2020.2989273

[3] S. Asif, Y. Wenhui, H. Jin and S. Jinhai, 'Classification of COVID-19 from Chest X-ray images using Deep Convolutional Neural Network', 2020 IEEE 6th International Conference on Computer and Communications (ICCC), 2020, pp. 426-433, doi: 10.1109/ICCC51575.2020.9344870

[4] Design of Accurate Classification of COVID-19 Disease in X-Ray Images Using Deep Learning Approach, Joy Iong-Zong Chen, DOI: https://doi.org/10.36548/jismac.2021.2.006

[5] D. Anand, D. Tank, H. Tibrewal and A. Sethi, 'Self-Supervision vs. Transfer Learning: Robust Biomedical Image Analysis Against Adversarial Attacks', 2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI), 2020, pp. 1159-1163, doi: 10.1109/ISBI45749.2020.9098369

[6] Nakao, T., Hanaoka, S., Nomura, Y. et al. Unsupervised Deep Anomaly Detection in Chest Radiographs. J Digit Imaging 34, 418–427 (2021). https://doi.org/10.1007/s10278-020-00413-2

[7] J. Zhang et al., 'Viral Pneumonia Screening on Chest X-Rays Us- ing Confidence-Aware Anomaly Detection', in IEEE Transactions on Medical Imaging, vol. 40, no. 3, pp. 879-890, March 2021, doi: 10.1109/TMI.2020.3040950

[8] Lee JG, Jun S, Cho YW, Lee H, Kim GB, Seo JB, Kim N. Deep Learning in Medical Imaging: General Overview. Korean J Radiol. 2017 Jul-Aug;18(4):570-584. doi: 10.3348/kjr.2017.18.4.570. Epub 2017 May19. PMID: 28670152; PMCID: PMC5447633.

[9] Swarup Kr Ghosh, Anupam Ghosh. "ENResNet: A novel residual neural network for chest X-ray enhancement based COVID-19 detection", Biomedical Signal Processing and Control, 2022

[10] Zhang, Alan. (2020). Covid-19 Chest X-ray Images: Lung Segmentation and Diagnosis using Neural Networks. International Journal on Compu- tational Science & Applications. 10. 1-11. 10.5121/ijcsa.2020.10501.

[11] Ahmed, S., Hossain, T., Hoque, O.B. et al. Automated COVID-19 Detection from Chest X-Ray Images: A High-Resolution Net- work (HRNet) Approach. SN COMPUT. SCI. 2, 294 (2021). https://doi.org/10.1007/s42979-021-00690-w

[12] Saood, A., Hatem, I. COVID-19 lung CT image segmentation using deep learning methods: U-Net versus SegNet. BMC Med Imaging 21, 19 (2021). https://doi.org/10.1186/s12880-020-0