Comparing Transfer Learning methods to classify Pneumonia in Chest X-rays using Grad-CAMs

Vedant Devank Patadia
1160482
Dept. of Computer Science
Lakehead University
Thunderbay, ON
vpatadia@lakeheadu.ca

Manan Ashok Kumar Patel
1160481
Dept. of Computer Science
Lakehead University
Thunderbay, ON
mpatel94@lakeheadu.ca

***Abstract*— The rapid evolution of deep learning applications in the medical field is proven to be a significant assistance to doctors, especially during the outbreak of COVID-19 when the differential on pneumonia proved to be a major testing ground for the diagnosis. As such, the availability of datasets with regard to pneumonia diagnosis through Chest X-Rays had been at surge, and so has the analytical research on them. One of the major research has been in the field of Transfer Learning, as the data is still in the nascent phase as progression on several different variables like age, gender, orientation, structure, type of pneumonia, etc are in preliminaries. Transfer Learning provides a way around those variables and helps directly differentiating the images as the features are already learnt, and better if the model in question is already trained on a similar domain. In our case, we are focusing on comparing short-term training on 3 models utilizing a TL method from a dataset from on a different domain. One subset of analytical research in this approach is the impact of augmentation on results. As the datasets are ever-evolving, our approach will make use of a small dataset with targeted augmentation and activation map visuals to help criticize a model evaluation, precisely how the model thinks. Generally, a trained model is interpreted using traditional methods like accuracy, F1 Score, precision, or recall but the overall idea behind a prediction still remains a probability score, for which a Grad-CAM[1] will prove to be an alternative. This will make it easier for a doctor who has never used such an application before to comprehend the underlying mechanism. In brief, we formulated an applicable pipeline for 1) Preliminary classification of data the models proposed without any augmentations, 2) Generating activation map visuals for the testing data 3) Employing targeted augmentations to models and finally 4) Discuss how well does transfer learning apply from a dataset like imageNet to the problem of Chest X-Rays and what affect does augmentation have on it.**

I. Introduction

For a few years, deep learning architectures have been used extensively in the field of clinical diagnosis. They have proven to be helpful to the fullest, especially during an overload of requests, but at the base, the overall process still needs human supervision, and a lot of suspicions are still present on the implementations. One of the main concerns is how a machine can learn in even a month compared to a professional who has trained for years. With the trends of technological advancements, even doctors are trying to adapt to those solutions. One of the main objectives of the project is to simplify the implementation process for anyone considering it and to give those who use it additional details about the model's approach to the data (through Grad CAM).

Pneumonia essentially refers to a lung infection that primarily affects the lungs' air sacs commonly known as alveoli [2]. According to Johns Hopkins, it is an acute infection in which alveoli fill up with pus or any other liquid[3]. Experts from American Lung Association say that symptoms of pneumonia may include cough, shortness of breath, fever, and shallow breathing [4]. According to the World Health Organization, in 2019, pneumonia alone is responsible for 14% of all deaths of children under 5 years old [5]. While observing Chest X-rays of patients suffering from pneumonia, a cloudy appearance may appear in segments in the lungs or even like patches solely dependent on the airspaces involved inside the lungs, there are a couple of other complications involved in the identification of pneumonia but as referred under [6], [7], there might be an increase in lung volume but never an opposite.

The main problem is that no two cases are exactly alike, and even the same cases change over time. Of course, it is outside the scope of our investigation to lay out facts or offer medical diagnosis intuitions, but what we are looking into will shed light on one of the training paradigms known as transfer learning. In particular, the speed in terms of iterations (epochs to be precise) and model inference based on the visual bias (GradCAM [1] Heatmap in our case).

Transfer Learning is a technique for adjusting a particular model that was previously trained on a different dataset[8]. Simply put, the features learnt before are transferred onto the primary set. Most of the time, transferring from a domain that is similar to the primary domain enables the model to be adjusted to the primary dataset. But in our case, obtaining weights for models that were pretrained on a sizable Chest X Ray database is either nearly impossible, comes with conditions, or is not trustworthy enough for short-term tuning. Thus, we go for one of the more widely available datasets, which here is imagenet[9], but one problem here is that the target set for the problem at hand is entirely different from the one we have in imagenet[9]. The basic instinct behind the idea is two parted, one that it will act as a high penalty ground but on second thought pretrained model will contain

some lower-level features already optimized. Thus, depending on the random initialization of top layer weights we can see some quick learning saturation.

A. Algorithms

After researching on algorithms like include VGG16, ZFNet, MobileNet_V2, Xception, Inception_V3, Inception_ResNet_V2, Inception_V4, ResNet50, ResneXt50 and DenseNet201. We trained VGG16, Resnet50, and DenseNet201 models to carry out comparative analysis for the binary classification of Chest X-Ray Images primarily because of the standardized usage over majority of Computer Vision applications and the high accuracy they generally provide.

- *VGG16*
  First proposed in 2014 by [10], Visual Geometry Group commonly known as VGG is a convolution Neural Network architecture ranging from 11-19 layers that stacks more layers onto AlexNet, and instead of having large hyperparameters they use smaller 3*3 size kernels in ConV Layer, and 2*2 in Max Pooling Layers [11]–[13]. VGG16 has 13 convolution layers with kernel size 3×3, followed by 3 fully connected layers being one of the deepest networks with 138 parameters [11], [12]. Further the versatility of the network is unmatched, and almost all the datasets or even researches on model architecture use it to set a benchmark or baseline.
- *ResNet50*
  Deep residual networks, such as ResNet50 [14], is a type of convolutional neural network used for image classification. The main innovation is the launch of the novel network-in-network design using residual layers [13], [15]. The major intention behind its inception was to increase the depth of the layer, where usually the accuracy decreases or saturates in a normal CNN structure. Resnet introduces, Skip connections, also known as shortcut connections or residuals, were used to tackle this issue when developing deeper models[11], [12]. ResNet50 takes images with dimensions up to 224*224 and has 50 residual networks.
- *DenseNet201*
  A contemporary CNN design called DenseNet [16], which was unveiled in 2017 requires fewer parameters for visual object detection. The output of a subsequent layer is combined with the result of the preceding layer [17]. DenseNet uses skip connections between blocks but dense connections between all of the layers within blocks to recognize visual objects[18]. It establishes feed-forward connections between each layer and every other layer. DensNet201 has L(L+1)/ 2 direct connections as opposed to standard convolutional networks with L layers and L connections [13].

B. Dataset

For our experiments, "Labeled Optical Coherence Tomography (OCT) and Chest X-Ray Images for Classification"[19] first by Daniel Kermany, Kang Zhang, and Michael Goldbaumin 2017 also available on Kaggle [20], is a great fit as the dataset is small as it focuses on children below 5 years of age and gives room for grooming it for the transfer learning models. The original dataset consists of 5856 frontal X-Ray JPEG images. The dataset is divided into 2 parts training, and testing. Each part is further categorized into 2 categories; *Pneumonia* (patients having either viral or bacterial pneumonia) and *Normal* (patients without any abnormalities).

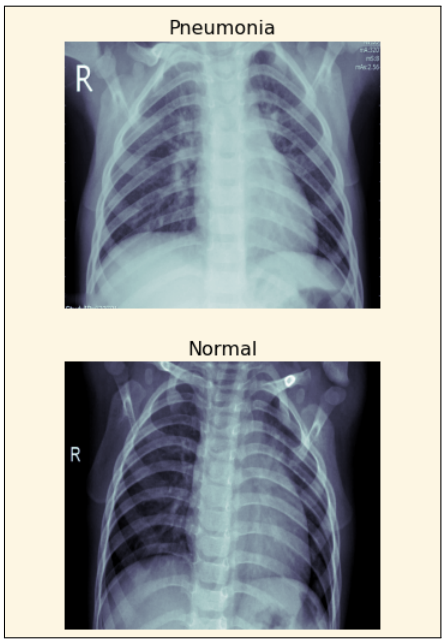


Fig. 1: Random Sample Image

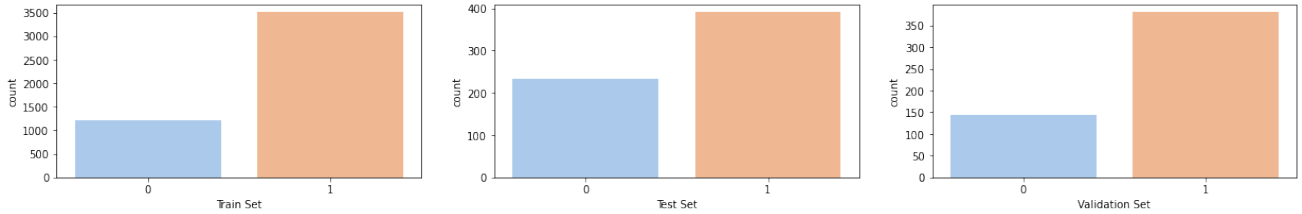
Chest X-Ray Images were chosen from retrospective batches of children patients aged one to five at the Guangzhou Women and Children's Medical Center in Guangzhou. All CXR imaging was done as part of the regular clinical treatment provided to patients.

All chest radiographs were initially checked for quality control before being removed from the study of the CXR pictures. Before the diagnosis for the photos could be used to train the AI system, they were graded by two experienced doctors. A third expert also reviewed the evaluation set to make sure there were no grading mistakes.

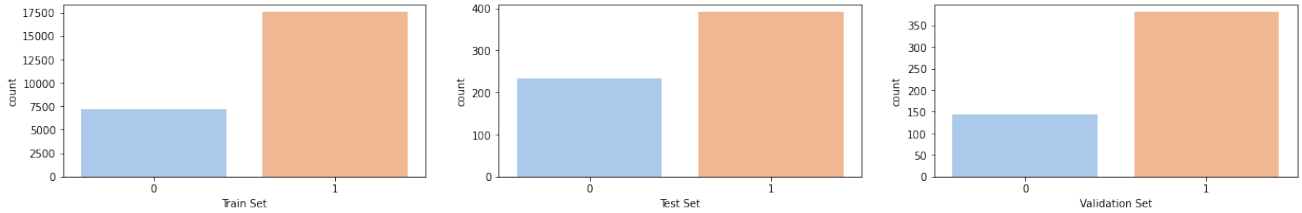
Looking at the data itself, the scans do have quite a few unnecessary details, like orientation indicators like letter "R" written on them as displayed in Figure 1.

C. Augmentations

While the dataset chosen is quite ideal considering it divides into viral pneumonia, bacterial pneumonia and normal Chest X-Ray images in a balanced fashion. But if we consider only Pneumonia vs Normal there is a big imbalance considering we are now merging balanced data to one side, thus giving rise to imbalance.



(a) Original dataset without augmentation (**Note:** 0 means Normal and 1 means Pneumonia)



(b) Modified dataset with augmentation

Fig. 2: Class Imbalance

To overcome this, we applied different augmentations to artificially inflate the existing datasets as proposed by [21] in their survey. On a closer note, the scans do have a lot of unnecessary details as well as they themselves have a variety in the image dynamics, such as brightness or contrast that can be used to our advantage. As such the augmentations that are available to us are,

- *Geometric Transformations*
  Geometric Transformations are easy to implement but the impact is quite acute considering the images already contain only singular styled representation, i.e., human and the images are having slight variations in terms of geometry, i.e., there are some variations in the angle of the positioned subject thus the model may already be robust to the simple transformations. Although warping may help in this matter as the records, even if taken from several different stations are still within the standards of scanning, thus variations in distance or parallax may help.
- *Flipping*
  As all the scans are taken in the same manner with only little deviations we can safely say that random flipping can impact the images, although the scale of such impact can be measured by simply feeding a trained model and inspecting the confusion matrix.
- *Cropping*
  Since cropping may help in reducing the unnecessary areas it may help but only in an acute fashion as Deep Learning techniques are already robust to noise and efficient in targeting parts of images necessary to them. Although image cropping can help in reducing the model stress.

Also, the pretrained model is trained on ImageNet, to the best weights to high accuracy, including different augmentations, thus giving a better ground to converge. To test this, we applied augmentations to the train set while targeting the "Normal" class to resolve the 1:3 approximated ratio and bring it closer to 1:2, which can be referenced in Figure 2b and then feeding it to train the deep learning models. Thus, the models are anticipated to overfit as they all try to resolve the problem of Vanishing Gradient while increasing their learning capacity.

II. Related Work

A. Previous Studies on Chest X-Ray

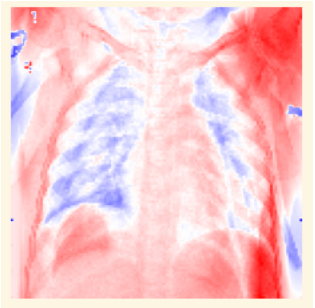
The rapid evolution of deep learning in clinical diagnosis has inspired many connoisseurs to outperform the existing algorithms in same domain. Such a feat was accomplished by [22] where they achieved state-of-the-art results in detecting 14 different diseases using the largest publicly available dataset, ChestX-ray14 [23]. An exceptional approach was used by [24] *et al.* where they implemented AlexNet and VGG16 models but they replaced Softmax Layer with SVM classifier to detect 12 different lung diseases on the same dataset used by [22]. Furthermore, Ashitosh *et al.* [25] performed a detailed comparative study of more than 20 articles based on various factors like data processing techniques, algorithms used, dataset, strengths, weakness, detection of different lung disease like lung cancer, pneumonia, tuberculosis. Authors from [26], implemented a 2 stage research where the first stage focused on detecting pneumonia using Modified AlexNet and second phase involves learning the features from stage 2 to improve classification accuracy during lung cancer evaluation.

B. Transfer Learning

The research combining Radiology and Deep Learning has been going on well before COVID-19 outbreak and has been quite concrete and focusing on the domain of Deep Learning the method is divided into 4 major structures, Model Curation, Transfer Learning, Comparative studies or Surveys and Additive or Adaptations. Here our interests lay in Adaptation and Transfer learning. Many research workers have tried to implement transfer learning on the OCT dataset and/or COVID dataset and implement in Chest X-Ray dataset. For instance,

(a) Average Normal

(b) Average Pneumonia



(c) Contrast Mean Images

Fig. 3: Image Averages and Differences

Daniel *et al.* [27] introduced transfer learning to make a generalized algorithm that classified images for macular degeneration and diabetic retinopathy whilst also distinguishing bacterial and viral pneumonia. Samir *et al.* [28] performed transfer learning in 3 parts making learning more stable and quicker. First, it primarily focuses on evaluating the effect of classification layer size on the final classification accuracy. Second, aims to evaluate how many layers should be frozen and the last part fine-tunes other parameters based on the previous best model. In [29], authors used Transfer Learning to combine the predictions using a weighted classifier and found an increase of 0.98% in testing accuracy.

C. Ensemble Learning

Different approach of ensemble learning was implemented by authors in [30], where they combine diverse extracted features from chest radiographs to perform classification. Tatiana [31] implemented RetinaNet, a 1 stage object detection model coupled with heavy augmentations and ResNext101 encoders, originally pre-trained on ImageNet dataset [32] for detecting pneumonia regions. Another pioneering strategy was introduced by authors from [33] implementing Dempster–Shafer theory, fusing 5 pre-trained CNNs in detection of pneumonia using CXR images.

D. GradCAM

Ramprasaath et. al.[1] proposed "Grad-CAM", a method that uses gradient of any target concept flowing into final convolution layer producing a localization map highlighting vital regions that help in prediction of the image. Dissimilar to any previous visualization technique, Grad-CAM can be used with a wide variety of deep learning models like CNN families, ResNet families and many more. [34] *et al.* focused on the effect of removing image background in pneumonia detection and provide plausible reasons with the help of GradCAM heatmaps.

The rest of this document is structured as follows: Section III addresses a brief step-by-step description of the methods used to achieve results. Section IV explains the model performance in touch with different parameter settings. Section V expresses the challenges faced during the tenure of this paper and which we were not able to overcome coupled with a concise discussion. This section is followed by Section VI, where we conclude our paper and pilot future enthusiasts to carry on more work in this field. Section VII acknowledges all who have motivated us to complete this paper followed by References.

III. Proposed Methodology

Our major interest lies in the generated GradCAMs and question whether they are beneficial to the comparisons. But, through the process strictly side with the deep learning part of the process and not comment on the medical diagnosis or prognosis.

Firstly, we have three models to compare, VGG16, ResNet50, and DenseNet201. The experiments are branched into 2 scenarios, Scenario 1 with no augmntations involved and Scenario 2 with augmentations which are discussed later on. Then expanding to the input sizes of 256*256 and 512*512 for each Scenario. Since, we're transferring weights from ImageNet, we have maintained a minimum batch size of 16 to carry out the training of the adapted binary classifiers on Chest X-Ray (CXR) Images.

1) *Scenario 1:* Considering the original dataset and predicting if the image has pneumonia or not with the plausible justification from the model Grad CAM.
2) *Scenario 2:* Applying a series of augmentations (discussed later) to the training set and the making predicting if the image has pneumonia or not with the plausible justification from the model Grad CAM.

The first scenario is quite straight forward but in the second one, we tried to address the class imbalance even if a little bit. In any case, the base starts with exploring data and verifying our options and set variables.

A. Preliminary EDA

Since the dataset only consists of images it is very crucial to understand the image properties such as aspect ratio etc. Suggestions for preliminary EDA were taken from [35]. During the EDA, we observed that the image size distribution and aspect ratio follow a right-skewed bell curve with most of the images having 1MB in size and aspect ratio approximately having a golden ratio of 1.6.

Further, we calculated the average images in both classes shown in Figure 3a and Figure 3b. Thus, to grasp a basic understanding of the difference between them, a contrast of mean is calculated in Figure 3c The mean simply shows us the probability of success that we can expect, which in this case the base of our data is very clear and it seems we don't need any cleanup and can dive directly to our base model building.

B. Transfer Learning and Model Building

Our Intention is quite simple, skip the step of building and training the model from scratch and dive directly to develop the pipeline that takes us to a Class Activation Map. Here, in transfer learning we have the model divided into two parts, one is the feature extractor which we carry forward from the original architectures while only changing the input shapes. The other part is called the top or the classifier which drives the predictions. Thus to train the model into our desired dataset, we keep the feature extractor frozen from training to keep the optimized learned features from the ImageNet dataset intact as otherwise they may get disturbed by the random initializer and high learning rate if the initial loss is too high.

Now in our case, the high-level features of both datasets involved in transfer learning are similar at best, so we need to train the feature extractor (even if partially) at some point to make a high-level change in the domain or else the huge feature extractor will only act as a data embedding mechanism which in our case is pretty much unreasonable. To, simplify the method interpretation that we introduced a new variable, saturation threshold. The threshold helps us in enabling the Feature Extractor, exactly its value corresponds to the point/epoch in training, at the end of which we unfreeze the Feature extractor layers so as to begin the fine tuning step.

Briefly, we train the model while keeping the feature extractor frozen, which is just coarsely training the classifier to align itself with the already optimized features, that is, until the saturation threshold, then after we enable training of the feature extractor, therefore going through the fine tuning of the model.

During our experiments, we assigned value of the threshold to be atleast 10 epochs, which is right after our first step decay of the learning rate. As it was observed that if the threshold is kept too low then the model erratically updates its weights thus leaving the structural learning of X-Rays quite difficult to achieve. But if the threshold is too high then the model will adapt to the embeddings from bottom and

1) Optimizers: In Deep Learning, Optimizer plays a vital role in improving the accuracy and reducing the loss of the model. Considering the survey done in [36], they found that 44% of previous studies have used Adam Optimizer and concluded that using Adam as as optimizer to train the model has improved the model performance by producing promising results. Considering the related studies we have implemented Adam optimizer to train our models.

2) Learning Rate: The initial value of learning rate is set to 0.001 with an added step scheduler to decrease Learning Rate on the end of every 10 epochs with a drop factor of 0.5 for a better curve and less reliance on random weight initialization of the top layers.

3) Batch Size: As a general norm, we started out with a batch size of 32, which is good until we add augmentations, on which Out Of Memory(OOM) Errors became quite prominent, especially during our experiments with the input size of 512.

So, we decided to reduce the batch size to 16 so that we can efficiently use the GPU memory without concern about OOM errors and maintain uniformity across the tests.

C. GradCAM Visualizations

Chest X-Rays superimposed with GradCAM are our proposed target of the pipeline, as they provide crucial information to criticize a particular prediction of a model. For example, a GradCAM being empty might mean the model is predicting the result on a random bias or fully highlighted GradCAM might mean the model is overly confident on the result but without proof or insight, it's not only helpful in training but can also provide more information than just 1s and 0s.

D. Applied Augmentations

We used *"albumentation"* library [37] to apply augmentations and artificially inflate our dataset. We performed more augmentation on the normal images (all augmentation from pneumonia images plus Horizontal Flip) to approximately balance the class imbalance as shown in Figure 2b. The list of augmentations that are performed on the normal images are as follows:

- *Rotation:* a random rotation ranging from 20° in clockwise as well as anti-clockwise direction.
- *Horizontal Flip:* making a horizontal flip of the image.
- *Contrast:* changing contrast with a factor range of 0.1.
- *Brightness:* changing brightness with a factor range of 0.1.
- *Hue & Saturation:* changing hue and saturation for an image with a factor range of 10 and 20 respectively.

Hyperparameters	**Values**
Optimizer	Adam
Learning Rate	0.001
Dropout	0.5

Augmentaion	**Value**
Rotation	± 20
Horizontal Flip	
Contrast	0.1
Brightness	0.1
Hue	10
Saturation	20

TABLE I: Hyperparameters Values and Applied Augmentation

IV. Experimental Results

We devised 2 scenarios, one with augmentation and one without it, along with 2 input sizes; 256*256 and 512*512, under three models with similar model parameters and same hyperparameters for support comparisons. The in the pipeline

Model	VGG16				ResNet50v2				Densenet201			
Scenario	1		2		1		2		1		2	
Input Size	256	512	256	512	256	512	256	512	256	512	256	512
Accuracy	0.71	0.69	**0.82**	**0.82**	0.73	0.72	0.74	0.76	0.73	0.76	0.75	0.76
Precision	0.62	0.59	**0.76**	**0.76**	0.64	0.63	0.66	0.69	0.64	0.69	0.67	0.69
Recall	0.84	0.82	**0.87**	**0.88**	0.84	0.82	0.83	0.86	0.84	0.86	0.84	0.85
F1-Score	0.60	0.56	**0.78**	**0.78**	0.64	0.62	0.66	0.69	0.62	0.69	0.67	0.69

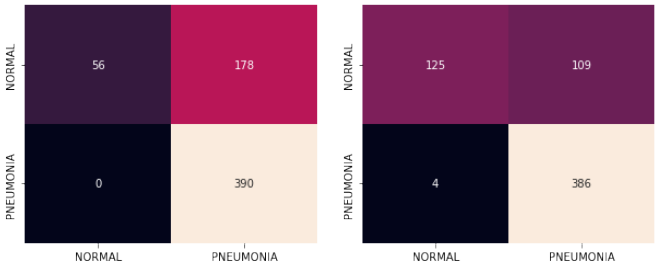
TABLE II: Model Result Comparion

we focus on the metrics as shown in Table II and then further compare using the Confusion Matrix. Thus on a per model conclusion we delve into the Training curves and the generated GradCAMs for training inspection and understanding the results respectively.

Before describing per model results, the best-performing model is VGG16 in Scenario 2 (with augmentation) and looking at the recall in Table II there is a very slight improvement in input shape of 512, while VGG16 is the best performer on input size of 256 on Scenario 2 as well.

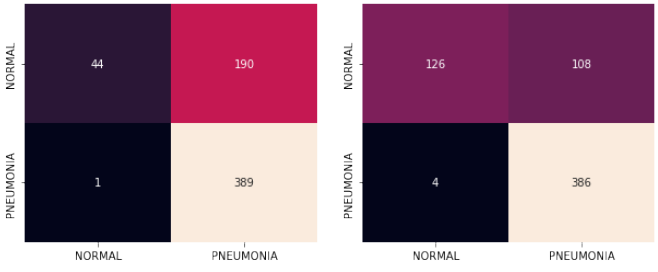
The worst performance is over VGG16 in Scenario 1 (without augmentation) on the input shape of 512. While the best-performing model on Scenario 1 is DenseNet201 with input size of 512, rather than input size of 256, ResNet50 barely moves up, looking at the F1-Score.

A. VGG16



(a) Scenario 1 **(b)** Scenario 2 (Augmented)

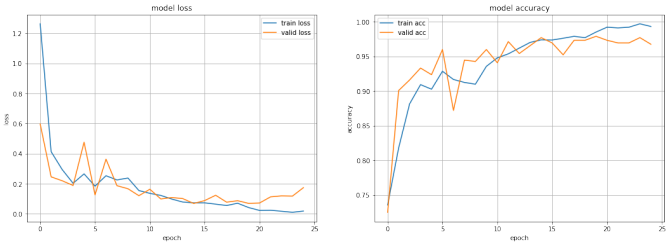
Fig. 4: Confusion Matrix of VGG16 with input size 256



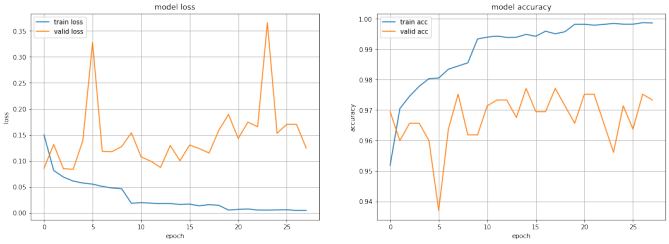
(a) Scenario 1 **(b)** Scenario 2 (Augmented)

Fig. 5: Confusion Matrix of VGG16 with input size 512

In the test concluding input size of 256, the diagonal of the confusion matrix reverts back in favour of the "NORMAL" class with the help of augmentations, which can be compared with the help of Figure 4a and Figure 4b. This is the reason we see an increase in accuracy as referred in Table II but at the same time we see that the increase in recall is not as high in magnitude as it seems for accuracy, which we see as the False Negative (FN) increase in Figure 4b.

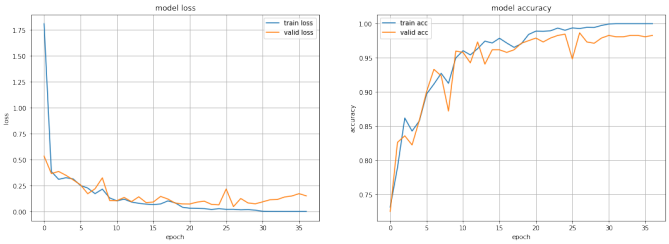


(a) Scenario 1

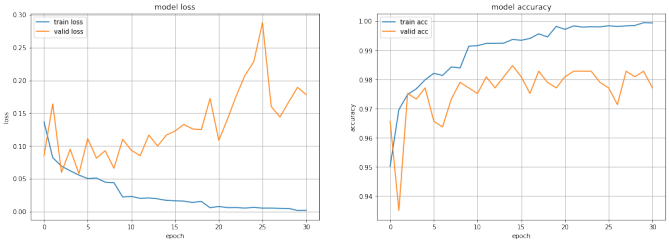


(b) Scenario 2 (Augmented)

Fig. 6: Model History of VGG16 with input size 256



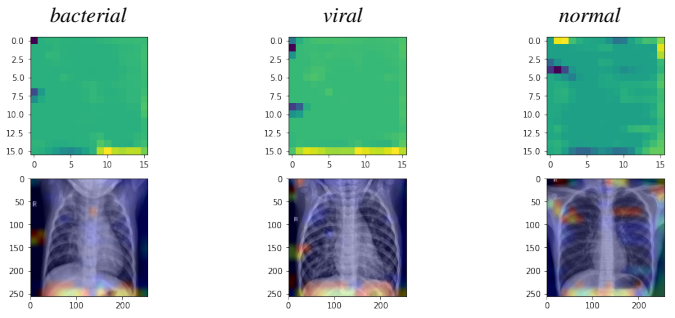
(a) Scenario 1



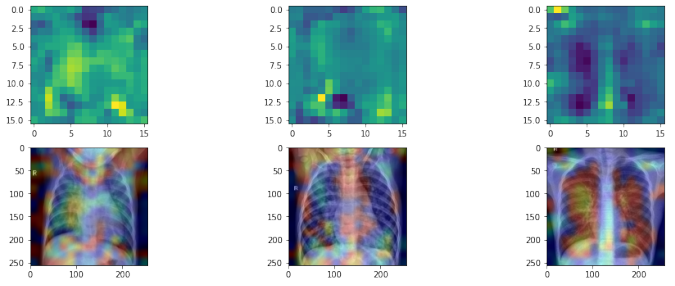
(b) Scenario 2 (Augmented)

Fig. 7: Model History of VGG16 with input size 512

Also, in the tests concluding the input size of 512, the results

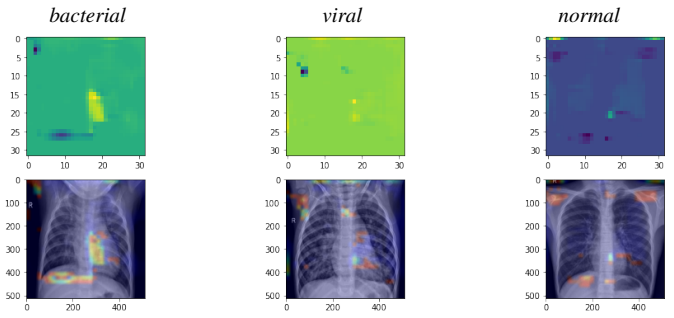


(a) Scenario 1

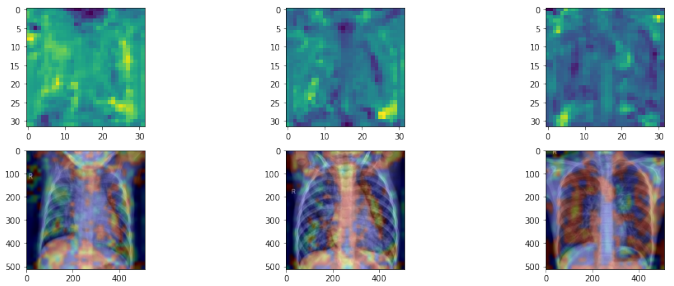


(b) Scenario 2 (Augmented)

Fig. 8: GradCAMs of VGG16 with input size 256



(a) Scenario 1



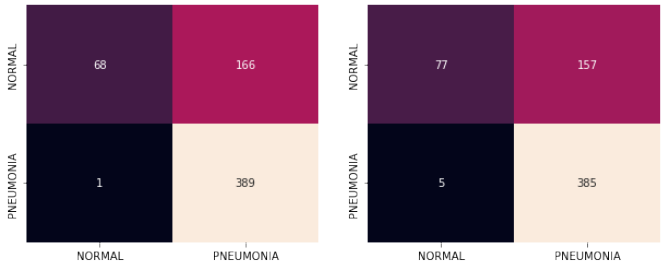
(b) Scenario 2 (Augmented)

Fig. 9: GradCAMs of VGG16 with input size 512

are almost as similar to that of results in input shape of 256, that is, better performance in tests including augmented data (Figures 5a& 5b). This is even true to the point of recall and increase in False Negative (FN) which can be seen in Figure 5b and Table II. Exploring more, the loss and accuracy curves, we observe an almost identical trend in comparing the input sizes, which can be inferred through Figures 6 & 7. Further comparing over the scenarios, the curves in Scenario 1 converge and remain close to each other, thus best epoch in terms of loss is bound to result in a generalized model. But in the case of Scenario 2, the curves diverge in either input size thus we cannot guarantee the model's generalization. While the result might be simply thought as a consequence of unbalanced classes and further by indirectly increasing the training iterations as well as the batch size using augmentations, to confirm this claim we overview the similarity between the input sizes of 256 and 512 in Figures 6 and 7 respectively.

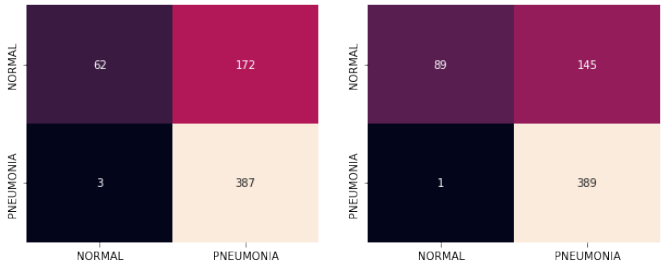
Simply, we can state that the chance of less generalized model may lead to a less preferred result. But we suggest that a further investigation of the GradCAM visuals of the classification layers can prove advantageous.

Thus, we move to Figures 8 and 9, in case of comparison to input sizes we see an increase in resolution and better structural learning of anatomy. Also, if comparing to the scenarios, we see a better visual representation of inference the model is generating. This can be the case of model memorizing the features of a skeleton but even if that is the case, within the best of 30 epochs it is still a better results on a minimal model build.



(a) Scenario 1 **(b)** Scenario 2 (Augmented)

Fig. 10: Confusion Matrix of Resnet50v2 with input size 256

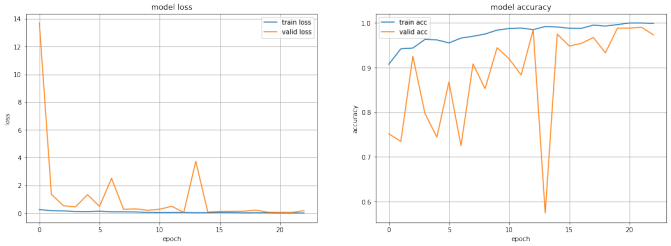


(a) Scenario 1 **(b)** Scenario 2 (Augmented)

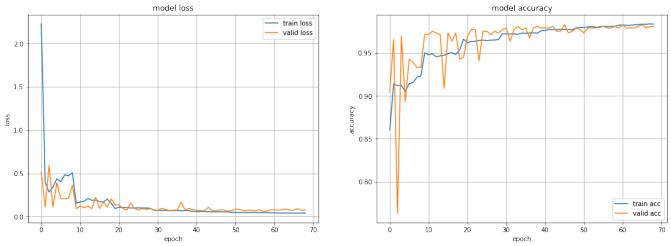
Fig. 11: Confusion Matrix of Resnet50v2 with input size 512

B. ResNet50v2

If compared to VGG16, the results are similar across all the tests and only a minor improvement while jumping in between scenarios, which can be understood using Table II where the if we focus on ResNet50 results over both the scenarios. This difference is quite comparable to the input size of 512, where

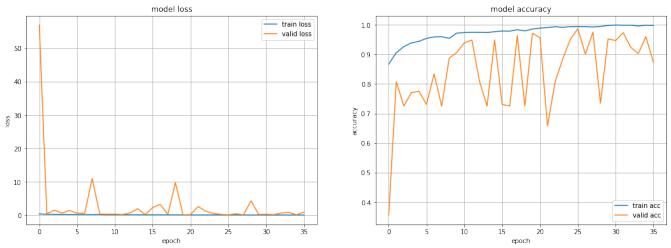


(a) Scenario 1

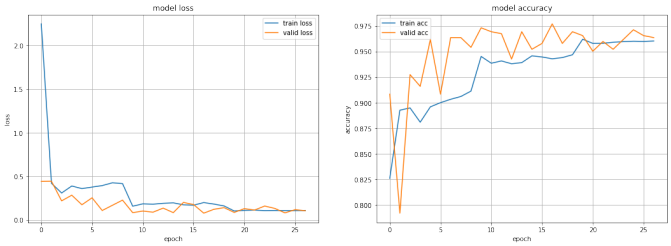


(b) Scenario 2 (Augmented)

Fig. 12: Model History of Resnet50v2 with input size 256

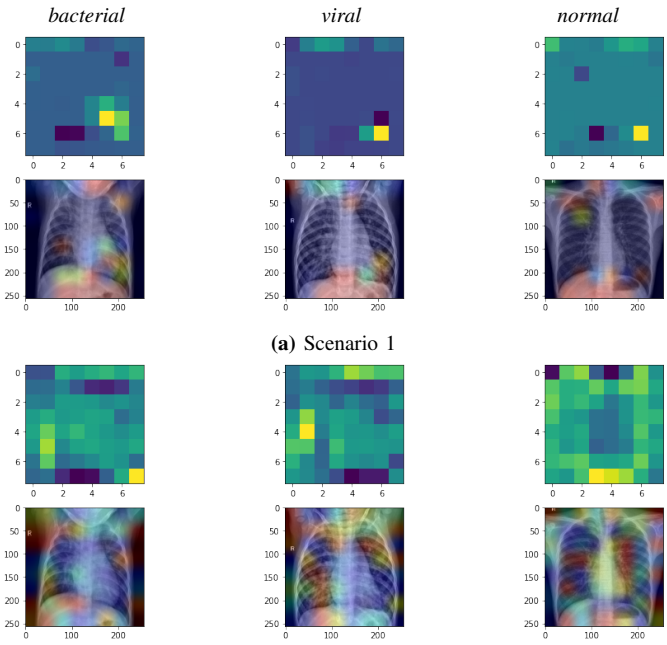


(a) Scenario 1



(b) Scenario 2 (Augmented)

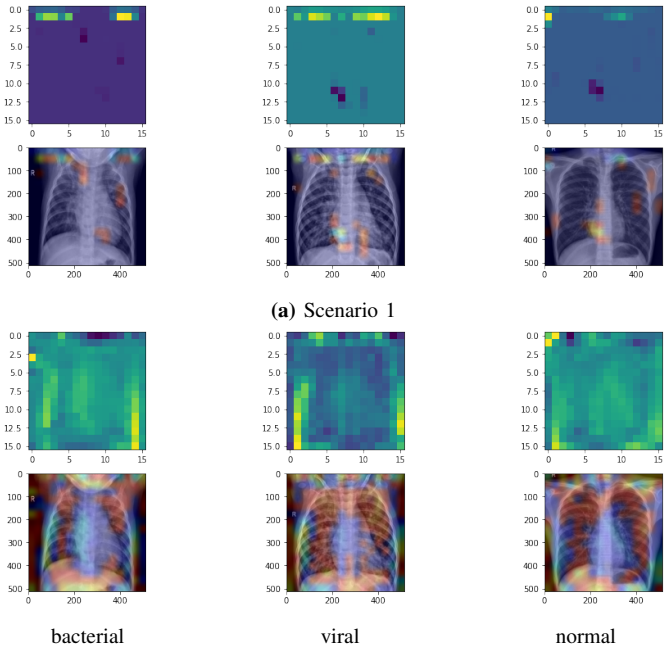
Fig. 13: Model History of Resnet50v2 with input size 512



(a) Scenario 1

(b) Scenario 2 (Augmented)

Fig. 14: GradCAMs of Resnet50v2 with input size 256



(a) Scenario 1

(b) Scenario 2 (Augmented)

Fig. 15: GradCAMs of Resnet50v2 with input size 512

we quantitatively say the results are better overall even if minor.

To better understand the numbers, we look at the confusion matrices of ResNet from the Figures 10 & 11 for the input sizes 256 and 512 respectively. While obviously the matrices diagonals are better looking and more plausible in VGG16 but internally there is a general increase with better data in both quantity and quality setup, precisely in comparing Figures 10a, 10b & 11a to Figure 11 on translating, more data through augmentations is ever so slightly improving the True Negatives (TN) which is also true in the case of improving the resolution, but only through augmentations, which is also evident in the overall decrease in False Negative(FN).

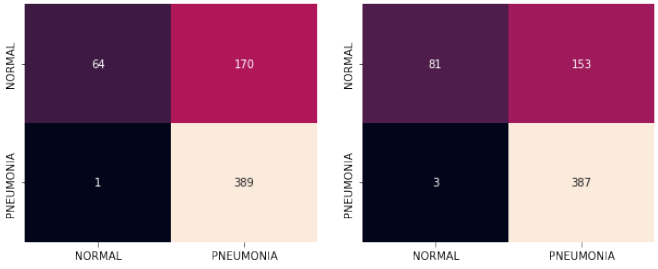
To further understand the performance, we can take a look at the Loss and accuracy curves in Figures 12 & 13, corresponding to their input sizes which surprisingly show quite general trend, as such the loss is quite high but the steep slingshot to lowest loss is literally in the first couple of epochs. Especially in Scenario 2 (Figures 12b & 13b), where they show quite a smooth curve unlike Scenario 1. In either of the four cases, the curves of validation and training converge on the loss curve, and we can say that the model ingests the

data in both scenarios very well and thus we can assume better generalization than VGG16. But looking back at the confusion matrices we say that due to high rate of False Positives the model performance suffers, in better words, above over 23% of data is incorrectly guessed by the model just because of False Positive results, which is also true to the case of VGG16 even though this time the loss curve seems quite promising.

The results now may correspond to the state, if the best possible loss on validation curve might not correspond to the best possible validation accuracy. Thus we move on to more data, that is the GradCAMs, and simply comparing Figures 14 & 15 through the input sizes we see an improvement in visual resolution over them, which is the same on comparing from Scenario 1 to Scenario 2 which is similar to the trend on VGG16. Although, in the case of ResNet we see that the model focus is more on the outline of object in question.

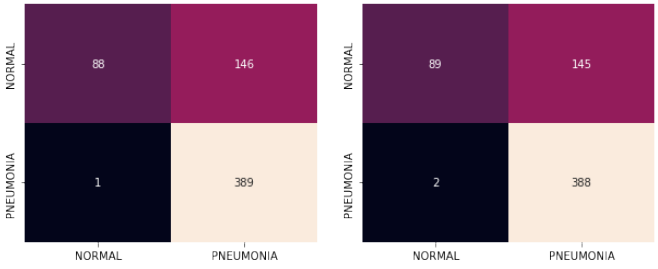
C. *DenseNet201*

The trend in DenseNet from Table II is quite similar to the step before, that is, there is a good improvement over the application of Augmentations to the pipeline, but only in the input size of 256.



(a) Scenario 1 **(b)** Scenario 2 (Augmented)

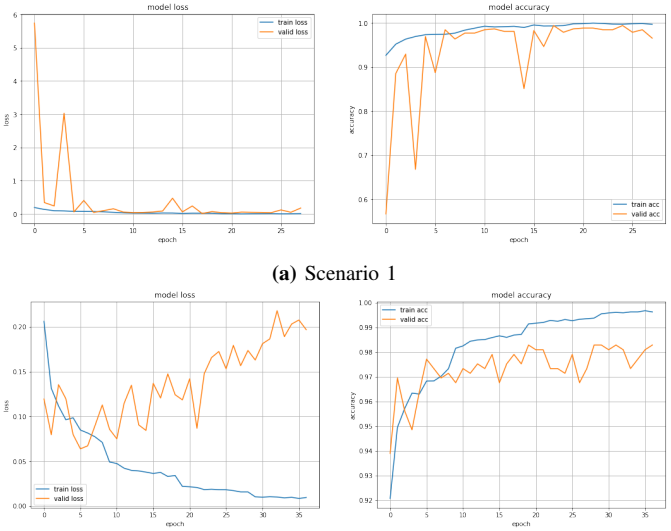
Fig. 16: Confusion Matrix of Densenet201 with input size 256



(a) Scenario 1 **(b)** Scenario 2 (Augmented)

Fig. 17: Model History of Densenet201 with input size 512

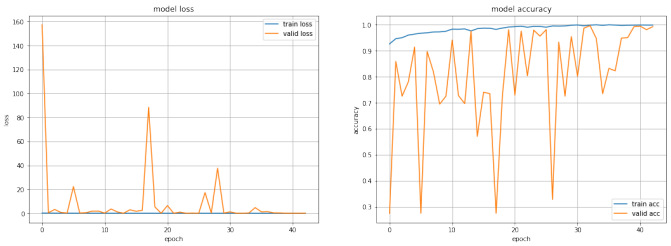
When increasing the input size to 512, the results are nearly identical in both scenarios. Looking closely, performance slightly improves over the increase in input size, over to the input size of 512 we see the saturation of what model can learn with the limits we have imposed. Even with augmentations as observed while moving from Figure 17a to Figure 17b the improvement is only in terms of single digits in True Negatives but we again see a single digit increase in False Negative thus balancing everything out. The only major improvement



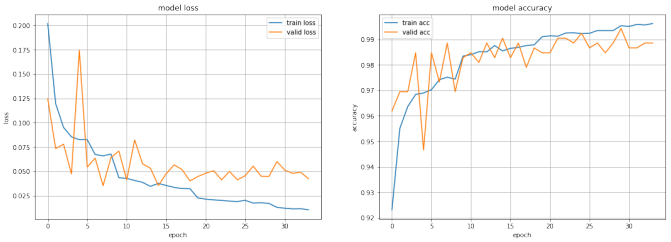
(a) Scenario 1

(b) Scenario 2 (Augmented)

Fig. 18: Model History of Densenet201 with input size 256



(a) Scenario 1

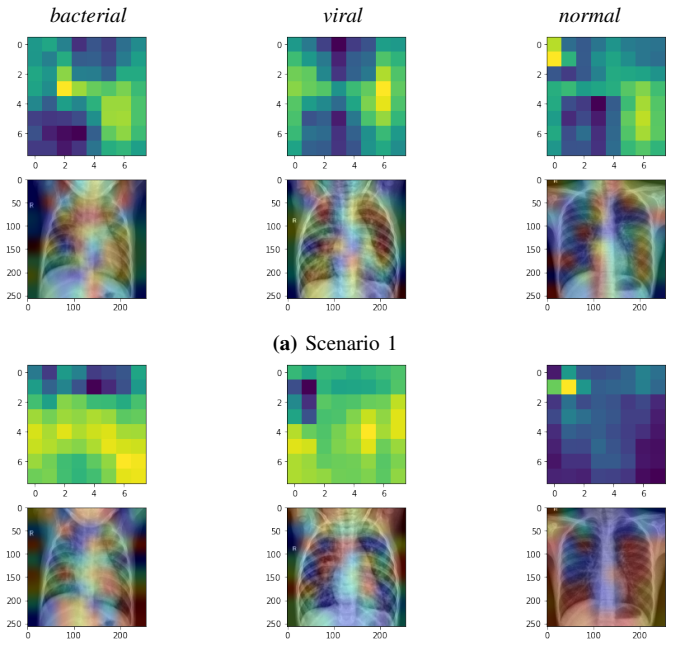


(b) Scenario 2 (Augmented)

Fig. 19: Model History of Densenet201 with input size 512

is in precision from Scenario 1 to Scenario 2 but the decrease in recall balances it out, thus we see a single digit bump in Accuracy.

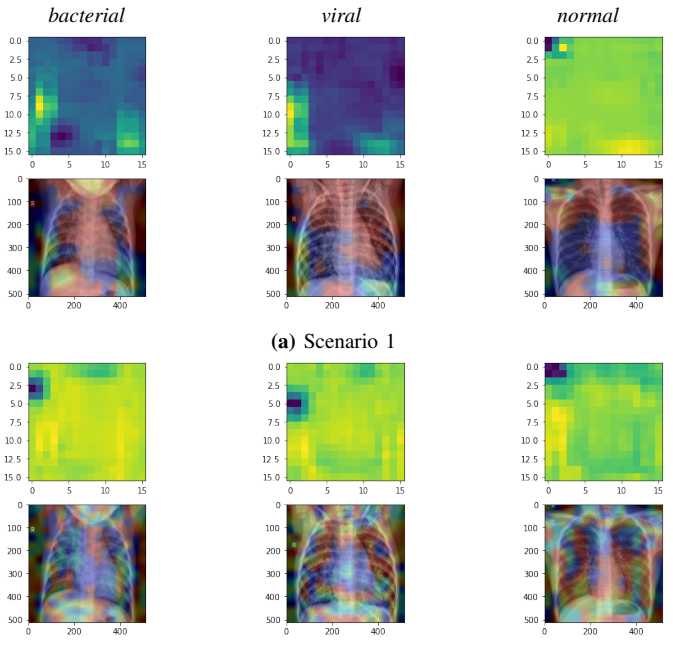
We can see the trend from ResNet50 transferring here, which is better seen in between the input sizes which can be better referred by moving from Figures 16a, 17a to Figure 17b. To investigate this further on the loss and accuracy curves on the Figures 18 & 13 corresponding to input sizes 265 and 512 respectively we can see learning saturation in Scenario 1 but a smoother graph in Scenario 2. Although, it may seem that the performance in Scenario 1 is better but that is solely due to the fact that the graph scale on y-axis is not the same in terms of loss displayed, this is to show the high spikes and better accommodate the whole. And talking about the transfer of trend from ResNet50, clearly adding augmentations translate



(a) Scenario 1

(b) Scenario 2 (Augmented)

Fig. 20: GradCAMs of Densenet201 with input size 256



(a) Scenario 1

(b) Scenario 2 (Augmented)

Fig. 21: GradCAMs of Densenet201 with input size 512

to better learning and bigger input size is leading to better convergence.

Adding to it, if we simply look at the model accuracy graphs in we can observe that the convergence is better in Scenario 1 for input size 256, which we can investigate in the GradCAM. With which we see that the confusion is present in generally all of the scenario, while of course confirming what we saw on the accuracy curve in Figure 18a. But if we look closely, there is a little more focus on outlines of object in Figure 21b.

One interesting observation here is even the base results on input size 512 on DenseNet (in Table II) are almost exactly similar, which is propagated through the confusion matrices in Figure 17 and the GradCAM in Figure 21

V. Discussion

Just by looking at the results we can say that the deep models like ResNet and DenseNet are heavily dependent on the amount and quality of data, while this is true for almost all the models with high capacity to learn but in our case this is very prominently seen in the models with higher depth. While this can even be due to the case of increasing model capacity in the feature extraction part but not proportionally doing the same in the classifier structure. The result of which is seen in, Table II, we observe that scenario 2 is slightly better than the scenario 1 in ResNet and DenseNet. But with VGG16 it was quite evident that augmentations play a role in improving performance, while we may not have catched the same results in ResNet and DenseNet but ever so slightly, in Figures 10a, 11a & 16b we can see improvements due to augmentations.

Our insight of this happening is due to the model depth and residual nature of ResNet and DenseNet, that they either failed on final tuning or the classifier capacity was not enough. This, of course, can be looked upon in the training loss curves. The loss curves on other hand show a different picture at the front. While, VGG and DenseNet clearly overfitting and even starting at quite a low magnitude of loss but looking closely and expanding on the axes we are able to see a trend and what can only be described as the effect of random weight initializer at play.

Before we move on to our discussions the interest areas on the X-Rays are not something we can easily comment upon. Although, the disease spread and basic domain knowledge is available to public reading, but the diagnostic knowledge needed to effectively comment upon intricate structures is not our in our expertise.

A. VGG16

The overall loss curve of VGG16 in Figures 6 & 7, is quite similar in both scenarios while the one with augmentations starts quite low while showing signs of underfitting (which is good). Also looking at around 10th epoch we can see a general descent, and quite a smooth one.

In VGG16, we can see the model is looking into generalization after it is given the dataset with augmentations which is quite evident in the superimposed images (bottom row) of the very last prediction (first from right), showing the interest areas around lungs. All in a very general improvement is shown already as looking at the results without augmentations are quite plain and the ones with augmentations show a slight structure.

B. ResNet50v2

The case of ResNet as referred in Figures 12 & 13 is almost similar to VGG but looking aside the validation loss we can see that the training is quite stable on the left without augmentation, whereas we can see a general curve with augmentations. Which shows us that in this case augmentation played a role in adding penalty to the regime. Although looking closely the model still tends to overfit in either case.

Furthermore, in the curve without augmentations show little to no fluctuations around epoch 10 on training curve but just a slight bump and then a bigger one on validation curve, which might just be because of learning rate change or random deviation, but looking back a similar trend is also shown in VGG16. But moving onto the one with augmentations, we can see a clear drop before the threshold (10) epoch, and which shows an overall improvement due to training of feature extractor.

Looking at the Grad-CAMs in ResNet, without augmentations the model is quite simply looking near the diaphragm and slightly above it while the other area seems quite irrelevant or simply deductions. But in the case of augmented data the model starts looking into the general structure of the human body and the interest areas are getting more generalized. Although in this case the effect of mid separation is not quite evident in heatmaps (mid row) but looking at the superimposed images (bottom row), it does seem that the model is targeting the lung areas for activations.

C. DenseNet201

DenseNet on another end shows a different trend on itself, without augmentations, the curve is as smooth with small but minor bumps on only validation curve similar to ResNet but no variations around threshold as shown in Figure 18 & 19.

In addition to that, the curve with augmentation is quite erratic on validation with no definitive bump after threshold epoch but a slight general descent is on the training curve after threshold. This is no clear indication of what to assume but a little more digging around is needed.

Now translating our curves onto the generated Grad-CAM, we see a difference in trend in the case of DenseNet, but one major aspect to keep in mind, our discussion will majorly depend on the mid row. Also, for reference, in each bounding box of Figures 8, 9, 14, 15, 20 & 21 showing Grad-CAMs, the first two are listed to be diagnosed with Pneumonia the very last column shows normal images.

The Grad-CAM visuals in DenseNet on other hand seem quite indifferent on a glance, but inferring from Table II, and Figures 20 and 21 they do seem to fare correctly. Either it be with or without augmentations, the training loss is already quite low which we can safely assume that due to the depth of the model structure and already known weights the learning process is quite slow or even stagnant. Although we can see structures around in the heatmaps but translating them on the X Rays, at most we can say that without augmentations there is an evident structure on lungs along with separation of spine. But the same doesn't transfer to the one with augmentations, which can be due to low penalty leading to less precision on short term training and incorrect threshold setup, the prior is already evident after observing the VGG16 and ResNet50, also can be confirmed by testing on DenseNet201. The latter on the other hand can be tested by training on different lengths but out of our scope.

VI. Conclusion and Future Scope

From the base we can see that the threshold we set on final tuning plays an important role in how the model slides back into a fit. Of course, our interpretation on X Rays as a medical reference is trivial but in conclusion to the training process, we can clearly see that without augmentation it would take more work, time and computational power to get a good fit however be the case of the top layers. Furthermore, we were able to see clear results on Grad-CAM that augmentations played a role in inducing the required penalty required to improve a model further, this is most dominant in VGG16 followed by ResNet and then DenseNet. The same is conclusive with the overall score with VGG16 being the best model thus.

Our results are quite limited to binary classification and simplistic top layer to decode vectors from feature extractor, also the efficacy of the results is relied upon random weight initialization. The future research can be done on, adding more classes like, ChestX-ray8[23], COVID-19 Pneumonia, Viral Pneumonia and Bacterial Pneumonia, and a more sophisticated top layer while improving efficacy with more reliable weight initializer may help build better training curves.

VII. Acknowledgements

We can't explain in words how grateful we are for Dr. Xing Tan's guidance and patience. He freely shared his knowledge and expertise, without which we would not have been able to embark on this path. Also, this project would not have been possible if the Acting Chair and Dean of Science and Environmental Studies of Lakehead University had not given us the opportunity.

Furthermore, we appreciate the assistance with error correction, late-night feedback sessions, and continuous encouragement from our classmates and cohort members, notably our schoolmates. We should also express gratitude to our coworkers, housemates, and anybody who has influenced and motivated us.

REFERENCES

[1] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," *International Journal of Computer Vision*, vol. 128, no. 2, pp. 336–359, Oct. 2019. DOI: 10.1007/s11263-019-01228-7. [Online]. Available: https://doi.org/10.1007/s11263-019-01228-7.

[2] "Learn about pneumonia — american lung association." (2022), [Online]. Available: https://www.lung.org/lung-health-diseases/lung-disease-lookup/pneumonia/learn-about-pneumonia.

[3] "Pneumonia — johns hopkins medicine." (), [Online]. Available: https://www.hopkinsmedicine.org/health/conditions-and-diseases/pneumonia.

[4] "Pneumonia symptoms and diagnosis — american lung association." (2022), [Online]. Available: https://www.lung.org/lung-health-diseases/lung-disease-lookup/pneumonia/symptoms-and-diagnosis.

[5] "Peumonia in children." (2022), [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/pneumonia.

[6] "X-ray atlas: Chest x-ray — glowm." (), [Online]. Available: https://www.glowm.com/atlas-page/atlasid/chestXray.html.

[7] H. Knipe. "Reticulonodular interstitial pattern — radiology reference article — radiopaedia.org." (2022), [Online]. Available: https://radiopaedia.org/articles/reticulonodular-interstitial-pattern.

[8] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Journal of Big Data*, vol. 3, no. 1, May 2016. DOI: 10.1186/s40537-016-0043-6. [Online]. Available: https://doi.org/10.1186/s40537-016-0043-6.

[9] O. Russakovsky, J. Deng, H. Su, *et al.*, *Imagenet large scale visual recognition challenge*, 2014. DOI: 10.48550/ARXIV.1409.0575. [Online]. Available: https://arxiv.org/abs/1409.0575.

[10] K. Simonyan and A. Zisserman, *Very deep convolutional networks for large-scale image recognition*, 2014. DOI: 10.48550/ARXIV.1409.1556. [Online]. Available: https://arxiv.org/abs/1409.1556.

[11] A. Uzila. "5 popular cnn architectures clearly explained and visualized." (2022), [Online]. Available: https://towardsdatascience.com/5-most-well-known-cnn-architectures-visualized-af76f1f0065e.

[12] R. Karim. "Illustrated: 10 cnn architectures." (2019), [Online]. Available: https://towardsdatascience.com/illustrated-10-cnn-architectures-95d78ace614d.

[13] K. E. Asnaoui, Y. Chawki, and A. Idri, *Automated methods for detection and classification pneumonia based on x-ray images using deep learning*, 2020. DOI: 10.48550/ARXIV.2003.14363. [Online]. Available: https://arxiv.org/abs/2003.14363.

[14] K. He, X. Zhang, S. Ren, and J. Sun, *Deep residual learning for image recognition*, 2015. DOI: 10.48550/ARXIV.1512.03385. [Online]. Available: https://arxiv.org/abs/1512.03385.

[15] K. E. Asnaoui, "Design ensemble deep learning model for pneumonia disease classification," *International Journal of Multimedia Information Retrieval*, vol. 10, no. 1, pp. 55–68, Feb. 2021. DOI: 10.1007/s13735-021-00204-7. [Online]. Available: https://doi.org/10.1007/s13735-021-00204-7.

[16] G. Huang, Z. Liu, L. V. D. Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jul. 2017. DOI: 10.1109/cvpr.2017.243. [Online]. Available: https://doi.org/10.1109/cvpr.2017.243.

[17] W. C. S. Low, J. H. Chuah, C. A. T. H. Tee, *et al.*, "An overview of deep learning techniques on chest x-ray and CT scan identification of COVID-19," *Computational and Mathematical Methods in Medicine*, vol. 2021, A. Jolfaei, Ed., pp. 1–17, Jun. 2021. DOI: 10.1155/2021/5528144. [Online]. Available: https://doi.org/10.1155/2021/5528144.

[18] E. Çallı, E. Sogancioglu, B. van Ginneken, K. G. van Leeuwen, and K. Murphy, "Deep learning for chest x-ray analysis: A survey," *Medical Image Analysis*, vol. 72, p. 102 125, Aug. 2021. DOI: 10.1016/j.media.2021.102125. [Online]. Available: https://doi.org/10.1016/j.media.2021.102125.

[19] D. Kermany, K. Zhang, and M. Goldbaum, *Labeled optical coherence tomography (oct) and chest x-ray images for classification*, 2018. DOI: 10.17632/RSCBJBR9SJ.2. [Online]. Available: https://data.mendeley.com/datasets/rscbjbr9sj/2.

[20] p. mooney paul. "Chest x-ray images (pneumonia)." (2018), [Online]. Available: https://www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia?datasetId=17810&sortBy=voteCount.

[21] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, Jul. 2019. DOI: 10.1186/s40537-019-0197-0. [Online]. Available: https://doi.org/10.1186/s40537-019-0197-0.

[22] P. Rajpurkar, J. Irvin, K. Zhu, *et al.*, *Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning*, 2017. DOI: 10.48550/ARXIV.1711.05225. [Online]. Available: https://arxiv.org/abs/1711.05225.

[23] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers, "Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases," 2017. DOI: 10.48550/ARXIV.1705.02315. [Online]. Available: https://arxiv.org/abs/1705.02315.

[24] K. Almezhghwi, S. Serte, and F. Al-Turjman, "Convolutional neural networks for the classification of chest x-rays in the IoT era," *Multimedia Tools and Applications*, vol. 80, no. 19, pp. 29 051–29 065, Jun. 2021. DOI: 10.1007/s11042-021-10907-y. [Online]. Available: https://doi.org/10.1007/s11042-021-10907-y.

[25] A. Tilve, S. Nayak, S. Vernekar, D. Turi, P. R. Shetgaonkar, and S. Aswale, "Pneumonia detection using deep learning approaches," in *2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE)*, IEEE, Feb. 2020. DOI: 10.1109/ic-etite47903.2020.152. [Online]. Available: https://doi.org/10.1109/ic-etite47903.2020.152.

[26] A. Bhandary, G. A. Prabhu, V. Rajinikanth, *et al.*, "Deep-learning framework to detect lung abnormality – a study with chest x-ray and lung CT scan images,"

Pattern Recognition Letters, vol. 129, pp. 271–278, Jan. 2020. DOI: 10.1016/j.patrec.2019.11.013. [Online]. Available: https://doi.org/10.1016/j.patrec.2019.11.013.

[27] D. S. Kermany, M. Goldbaum, W. Cai, *et al.*, "Identifying medical diagnoses and treatable diseases by image-based deep learning," *Cell*, vol. 172, no. 5, 1122–1131.e9, Feb. 2018. DOI: 10.1016/j.cell.2018.02.010. [Online]. Available: https://doi.org/10.1016/j.cell.2018.02.010.

[28] S. S. Yadav and S. M. Jadhav, "Deep convolutional neural network based medical image classification for disease diagnosis," *Journal of Big Data*, vol. 6, no. 1, Dec. 2019. DOI: 10.1186/s40537-019-0276-2. [Online]. Available: https://doi.org/10.1186/s40537-019-0276-2.

[29] M. F. Hashmi, S. Katiyar, A. G. Keskar, N. D. Bokde, and Z. W. Geem, "Efficient pneumonia detection in chest xray images using deep transfer learning," *Diagnostics*, vol. 10, no. 6, p. 417, Jun. 2020. DOI: 10.3390/diagnostics10060417. [Online]. Available: https://doi.org/10.3390/diagnostics10060417.

[30] M. Masud, A. K. Bairagi, A.-A. Nahid, *et al.*, "A pneumonia diagnosis scheme based on hybrid features extracted from chest radiographs using an ensemble learning algorithm," *Journal of Healthcare Engineering*, vol. 2021, D. Singh, Ed., pp. 1–11, Feb. 2021. DOI: 10.1155/2021/8862089. [Online]. Available: https://doi.org/10.1155/2021/8862089.

[31] T. Gabruseva, D. Poplavskiy, and A. A. Kalinin, "Deep learning for automatic pneumonia detection," 2020. DOI: 10.48550/ARXIV.2005.13899. [Online]. Available: https://arxiv.org/abs/2005.13899.

[32] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Jun. 2009. DOI: 10.1109/cvpr.2009.5206848. [Online]. Available: https://doi.org/10.1109/cvpr.2009.5206848.

[33] S. B. Atitallah, M. Driss, W. Boulila, A. Koubaa, and H. B. Ghézala, "Fusion of convolutional neural networks based on dempster–shafer theory for automatic pneumonia detection from chest x-ray images," *International Journal of Imaging Systems and Technology*, vol. 32, no. 2, pp. 658–672, Sep. 2021. DOI: 10.1002/ima.22653. [Online]. Available: https://doi.org/10.1002/ima.22653.

[34] Y. Yang, G. Mei, and F. Piccialli, "A deep learning approach considering image background for pneumonia identification using explainable AI (XAI)," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, pp. 1–12, 2022. DOI: 10.1109/tcbb.2022.3190265. [Online]. Available: https://doi.org/10.1109/tcbb.2022.3190265.

[35] E. Byeon. "Exploratory data analysis ideas for image classification." (2020), [Online]. Available: https://towardsdatascience.com/exploratory-data-analysis-ideas-for-image-classification-d3fc6bbfb2d2.

[36] D. Meedeniya, H. Kumarasinghe, S. Kolonne, C. Fernando, I. D. la Torre Dıez, and G. Marques, "Chest x-ray analysis empowered with deep learning: A systematic review," *Applied Soft Computing*, vol. 126, p. 109 319, Sep. 2022. DOI: 10.1016/j.asoc.2022.109319. [Online]. Available: https://doi.org/10.1016/j.asoc.2022.109319.

[37] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, and A. A. Kalinin, "Albumentations: Fast and flexible image augmentations," *Information*, vol. 11, no. 2, p. 125, Feb. 2020. DOI: 10.3390/info11020125. [Online]. Available: https://doi.org/10.3390/info11020125.