COVID-19 Image Classification

Shichao (Michael) Liang
Georgia Institute of Technology
sliang76@gatech.edu

Michael C. Hayes
Georgia Institute of Technology
mhayes64@gatech.edu

Abstract

In this paper, deep learning models were trained and tested on over 2,900 different X-ray images of patients with a variety of medical conditions in order to potentially identify patients with COVID-19. Since Covid-19 identification is often complicated by co-factors and co-morbities from other illnesses, a data-set with combined X-ray images from a COVID-19 database, X-ray images from healthy patients, and X-ray images from patients with viral pneumonia from a pneumonia database was collated in order to avoid an overly simplistic binary classification task. A number of modern convolutional neural network (CNN) architectures along with an extensive tuning of their hyper-parameters in order to achieve a maximum accuracy of 98.45%.

The accompanying code repository for this paper can be found at:
https://github.com/mliang1987/DL_F20_Project

1. Introduction & Motivation

Coronavirus disease 2019 (COVID-19) is a contagious respiratory and vascular disease that is responsible for a global pandemic in 2020. Rapid diagnosis and differentiation of COVID-19 from other similar respiratory illnesses is crucial to both treatment and prevention. The primary objective of this paper is to apply a variety of deep learning CNN architectures to X-ray images from healthy patients, viral pneumonia (VP) patients, and COVID-19 patients with the hopes that the results from this paper help future projects determine the most effective architecture for the task of COVID-19 X-ray image classification.

Since COVID-19 is so novel, there are very few significant academic publications using this particular COVID-19 data-set. Thus, there is potential that the images in this particular data-set can be useful to help train state-of-the-art classification algorithms to assist doctors for COVID-19 diagnosis. This project will be limited to using X-ray scans for classification; clinical diagnosis will often require physicians to adjudicate based on multi-modal input. Alternative data and models to assist the image classification task outlined in this paper can be found in the future works section.

2. Related Works & Differences

While academic publications based on this data-set are sparse, there are multiple attempts by machine learning practitioners on this classification task. Multiple notebooks have been posted on the corresponding Kaggle data-set link, but the notebooks do not appear to be very extensive. Typically, other projects pose this problem as a binary classification task with an equal number of X-rays of COVID-19 cases and healthy cases. In addition, most prior work is not comparative of neural network architectures. Additionally, most of these notebooks were completed a few months ago, which means the number of COVID-19 X-ray samples were sparse. Since the COVID-19 database used has continuously expanded, this project uses a larger number of COVID-19 X-rays on which to train and test. In addition, the data-set is augmented with other viral pneumonia X-ray scans, which represent more realistic use-cases for the information. In summary, this project differentiates the classification task in the following ways:

- Using a larger variety of models.
- Using the most recent COVID-19 X-ray images for the largest amount of possible data.
- Includes X-ray images of patients with viral pneumonia along with healthy patients.

3. Data

The data-set [5] contains three different types of X-ray images: images of patients with COVID-19, images of patients with viral pneumonia caused by a different virus, and images of healthy patients. This data was collected from different sources and were found in the following repositories:

- COVID-19 X-ray images are from the Italian Society of Medical and Interventional Radiology (SIRM) COVID-19 DATABASE, Novel Corona Virus 2019

Dataset developed by Joseph Paul Cohen and Paul Morrison and Lan Dao in GitHub and images extracted from 43 different publications. The Github repository can be found in reference [2].

- Healthy/Normal and Viral Pneumonia X-ray images were adopted from Chest X-Ray Images (pneumonia) database.

See Figure 1 for representative images of the three classes.

3.1. Data Preprocessing

All X-ray images were provided in a PNG file format with a resolution of 1024×1024 pixels. These images were converted to a resolution of 256×256 pixels; the reduced pixel resolution is required by popular libraries for CNN architectures.

Since the data-set is imbalanced with a low amount of COVID-19 X-ray images compared to the other classes, the data-set was down-sampled to force all classes to have the same number of samples, with random sampling from the two classes with more than 219 samples (219 is the number of COVID-19 X-ray images) so that all classes have the same number of samples as the COVID-19 class. Note that this means that accuracy is a trustworthy evaluation metric after down-sampling since the data-set is now perfectly balanced between the three classes.

3.2. Data Statistics

Since the data-set informs a multi-class classification problem with imbalanced data, understanding the distribution of the data is important. From the data-set, the three classes have counts as follows:

Table of Counts per Class	
Number of COVID-19 Cases (Class 0)	219
Number of Healthy Cases (Class 1)	1,341
Number of Viral Pneumonia Cases (Class 2)	1,345
Total Number of Cases	**2,905**

After down-sampling, all classes contain the same number of samples. This was achieved by randomly selecting 219 cases from each of the healthy patient image sets and viral pneumonia patient image sets, so that all classes contain exactly 219 samples. The following table shows the final amount of data for each class:

Counts per Class After Downsampling	
Number of COVID-19 Cases (Class 0)	219
Number of Healthy Cases (Class 1)	219
Number of Viral Pneumonia Cases (Class 2)	219
Total Number of Cases	**657**

3.3. Train / Test / Validation Split

After downsampling, the combined 657 X-ray images were split into train, test, and validation sets. First, the down-sampled data-set was split into train and test using an 80 / 20 split: 528 (176 for each class) images were retained while 129 (43 for each class) images are used for testing. Then, the retained set was further split into training and validation sets using a 70 / 30 split: 369 (123 for each class) images are used for training while and 159 (53 for each class) images are used for validation. Therefore, the amount of X-ray images per class for each of these three sets are as follows:

Final Counts per Class			
	COVID-19	Normal	Pneumonia
Training Set	123	123	123
Validation Set	53	53	53
Test Set	43	43	43
Total	**219**	**219**	**219**

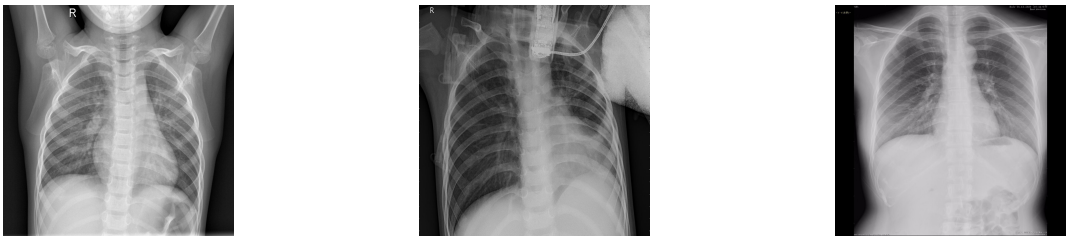
4. Methodology

Three different CNN models were trained on the data-set and tuned with a variety of hyper-parameters with the goal of maximizing accuracy: VGG16, Xception, and DenseNet121.

Very Deep Convolutional Networks (16-layers) from the Visual Geometry Group (VGG16) is a deep convolutional network with deeper layers and smaller filters [6] compared to its predecessor, AlexNet. In general, the 3×3 filters for VGG16 have the same receptive field as a 7×7 filters. Overall, the model is quite costly due to a large number of parameters, but is very performant on the ImageNet Large Scale Visual Recognition Challenge data-set.

The Xception model [1] is derived from the Inception model [7]. Whereas its predecessor embodied the idea of "going wide" with the network, the Xception model separates the spatial correlation mapping from the cross-channel correlation mapping. In essence, this is a "depthwise separable convolution", which is a depthwise convolution followed by a pointwise convolution. Intuitive, this 2D mapping followed by a 1D mapping is easier to learn than a full 3D mapping.

The DenseNet model [3] is inspired from the core idea of ResNets by sharing information between layers of different depths. In a traditional feed-forward network, an arbitrary layer is created from a non-linear function applied to the previous layer, or: $x_l = H_l(x_{l-1})$. DenseNet shares information between layers by considering data from all previous layers, or: $x_l = H_l(x_0, x_1, ..., x_{l-1})$. DenseNet takes a significant amount of memory to train since layer information cannot be discarded because it will be used in future layers, but higher accuracy can be achieved by these additional connections.



(a) Healthy Patient (b) Viral Pneumonia Patient (c) COVID-19 Patient

Figure 1: Representative X-rays scans from healthy patients, viral pneumonia patients, and COVID-19 patients.

For all models, transfer learning was leveraged in order to speed up learning, using the learned weights from models trained under ImageNet as a starting point. Additionally, each network model was slightly modified with the addition of a 1,024 dimensional fully-connected affine layer with a ReLU activation following another affine layer with 3 dimensions using a Softmax activation corresponding to the three classes for our data.

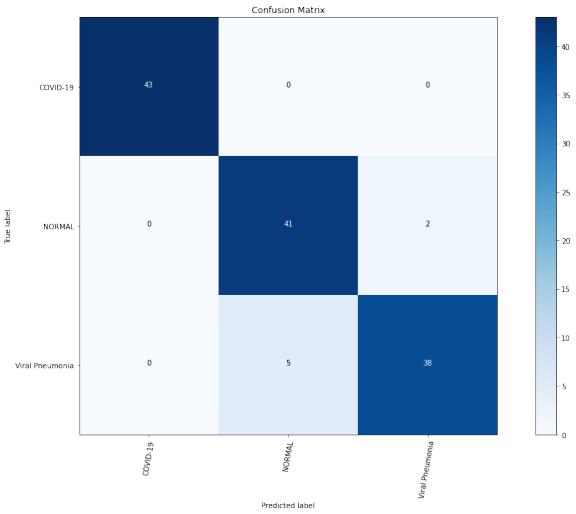
Attention was paid to the following hyper-parameters: learning rate, number of epochs, batch size, and different settings for the optimizer. The results in the following subsection show the hyper-parameters that yield the highest accuracy; which is the evaluation metric used to optimize the models. Furthermore, for each model, a final accuracy value, a confusion matrix for all classes based off the held-out testing data0set, train/validation loss diagrams, and train/validation accuracy diagrams are provided.

All models were created using Keras with a Tensorflow backend. The loss function used was the provided cross-entropy loss with respect to ground-truth labels. In addition, Adam optimizer was used for each model.

5. Results

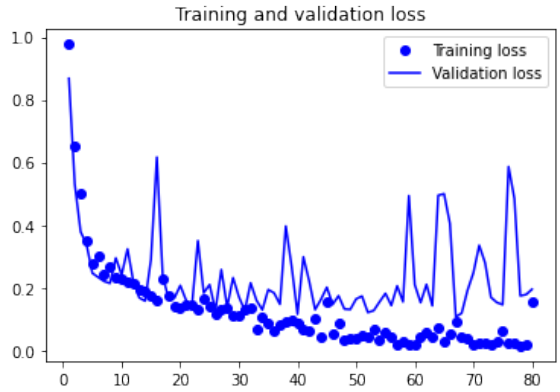
5.1. VGG16

For VGG16, the highest accuracy achieved was 94.57%. Note that this value can be recovered by summing the diagonal elements of the confusion matrix below divided by the total number of samples in the testing dataset (129). The confusion matrix corresponding to this accuracy is:

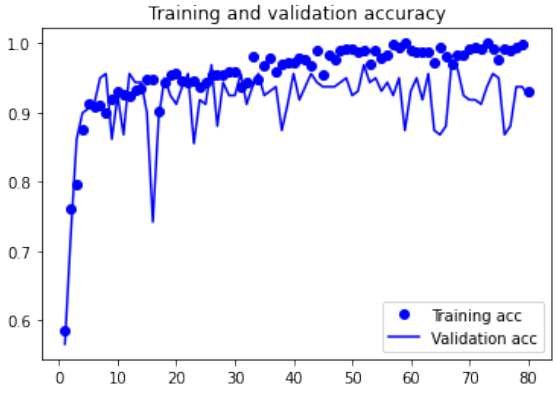


Note that the rows represent the true labels for each class and the columns represent the predicted labels for each class.

The training and validation loss over 80 epochs for VGG is as follows:



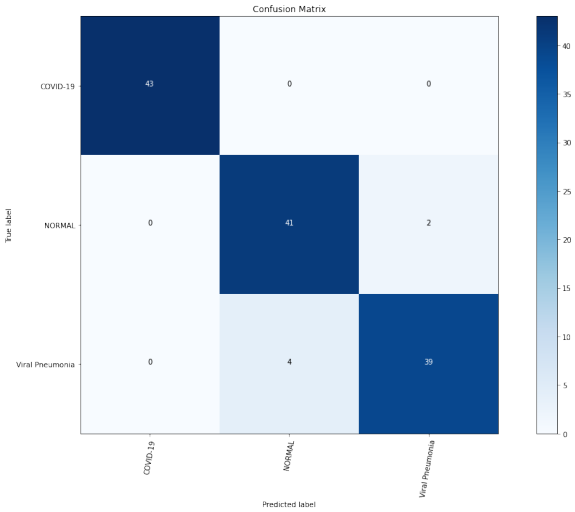
Finally, the accuracy over the same 80 epochs for VGG is as follows:



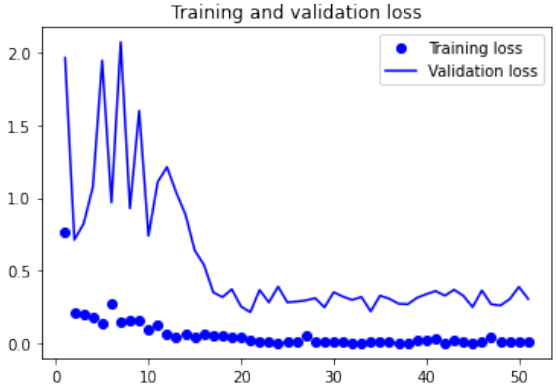
The hyperparameters that yield 94.57% accuracy were: a learning rate of $1e^{-5}$, a batch size of 10, training for 80 epochs, and using Adam as the optimizer.

5.2. Xception

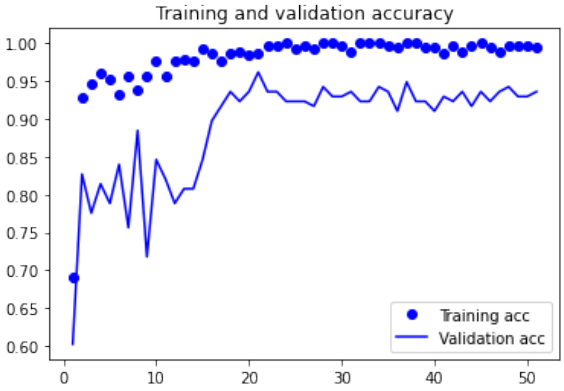
For Xception, the highest accuracy achieved was 95.35%. The confusion matrix corresponding to this accuracy is:



The training and validation loss over 60 epochs for Xception is as follows:



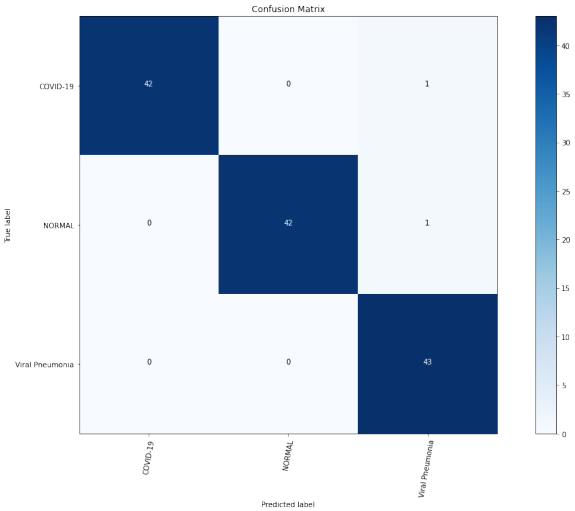
Finally, the accuracy over the same 60 epochs for Xception is as follows:



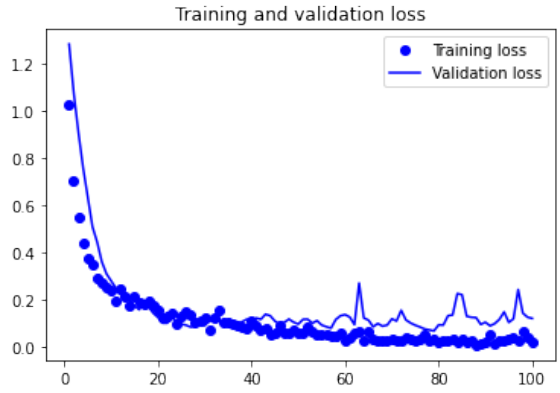
The hyper-parameters that yield 95.35% accuracy were: a learning rate of $5e^{-4}$, a batch size of 40, training for 60 epochs, and using Adam as the optimizer.

5.3. DenseNet121

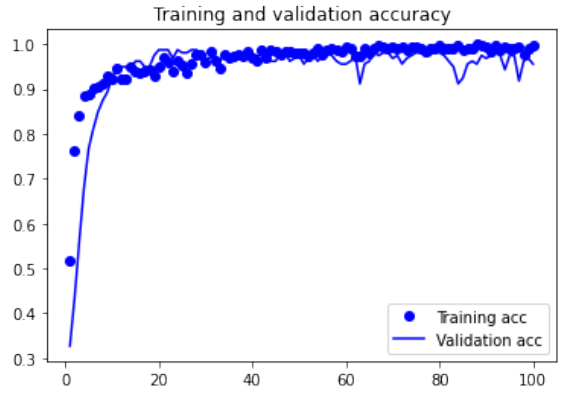
For DenseNet, the highest accuracy achieved was 98.45%. The confusion matrix corresponding to this accuracy is:



The training and validation loss over 100 epochs for DenseNet is as follows:



Finally, the accuracy over the same 100 epochs for DenseNet is as follows:



The hyperparameters that yield 98.45% accuracy were: a learning rate of $1e^{-5}$, a batch size of 10, training for 100 epochs, and using Adam as the optimizer.

6. Discussion

Overall, all 3 architectures demonstrated high accuracy for differentiating between the three classes, which was the project's metric for success. Generally, more modern architectures demonstrated better results, which is not surprising. While training hyper-parameters, the authors noted that the most difficult class to classify was viral pneumonia, with false positives appearing that should have been classified as healthy. However, differentiating between viral pneumonia and COVID-19 was fairly easy to learn. In terms of training metrics, while training loss was generally lower than validation loss, the overall overfitting was low, as both training and validation accuracy for all classes were close to the final testing accuracy. From this, we can infer that the models all generalized fairly well to this class of problem.

One of the biggest concerns with the current dataset is the number of COVID-19 X-ray images. In general, deep learning models require a significant number of data samples in order for the model to generalize well for the purposes of real world application. The current dataset only contains 219 COVID-19 X-ray images which is an insufficient amount of data for deep learning based applications. In particular, there exists significant difficulty in acquiring medical data; COVID-19 X-ray images and medical data in general is private and difficult to acquire. Additionally, COVID-19 is still a recent disease, meaning that there are very few extensive data collection projects for public use. However, as more COVID-19 data becomes available, the models can be retrained using additional data.

Transfer learning was of a huge benefit for all models, as the technique drastically reduced training time. Most models were able to train fully to convergence in under an hour on Google Colab, a cloud notebook service. It is also the opinion of at least one of the authors that the tutorials for Keras and Tensorflow result in an easier-to-understand and implement framework compared to PyTorch. Existing tutorial code was very easy to adapt to the classification problem at hand.

7. Future Works

As mentioned in the section above, as more data becomes available, models generated can generalize better for this classification task. In addition, future projects can also apply other forms of data and accompanying models to supplement X-ray image classification. Some examples of alternative data with their corresponding models are as follows:

- Patient information like age, gender, weight, or family history (genetics) can be used in a more traditional tabular data setting with alternative deep learning model to supplement diagnosis.
- A description of the patients' symptoms (coming from the patient's perspective) can be used as input data. A natural language processing model can be used to interpret the symptoms and output potential candidate diseases.
- More robust CT and X-Ray scan information in the form of the Digital Imaging and Communications in Medicine (DICOM) standard [4].
- Other routine data collected and used for medical diagnoses such as blood work results can be used as input data for other types of machine learning models.

The predictions from the different models listed above can be combined together with the image classification prediction outlined in this paper in order to create a more powerful (and likely more accurate) predictive model.

Finally, since differentiating between healthy and sick scans seemed to be the predominant issue, more work could be done to cascade two different models: One model to binary classify X-ray scans of patients with an illness with X-ray scans of healthy patients, and then subsequent classifier model of the type of illness based on the results of the first model.

References

[1] François Chollet. Xception: Deep learning with depthwise separable convolutions, 2017.

[2] Joseph Paul Cohen, Paul Morrison, and Lan Dao. *Covid Chest X-ray Dataset*, accessed November 12, 2020.

[3] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks, 2018.

[4] Oleg S. Pianykh. *Digital Imaging and Communications in Medicine (DICOM): A Practical Introduction and Survival Guide*. Springer Publishing Company, Incorporated, 2011.

[5] Tawsifur Rahman. Covid-19 radiography database. Dataset on Kaggle. https://www.kaggle.com/tawsifurrahman/covid19-radiography-database.

[6] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2015.

[7] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions, 2014.