Detection of COVID-19 using Convolutional Neural Networks

Dhruv Sanchety, Cody Swain

***Abstract*—Efficient detection of the COVID-19 infection is critical to make informed decisions that can prevent further infections and deaths. The paper designs, implements, and evaluates a multi-class convolutional neural network (CNN) architecture for the efficient identification of the COVID-19 infection. Analysis of X-Ray images that are labelled are utilized in the evaluation of the proposed technique.**

***Index Terms*—Deep Learning, Convolutional Neural Networks (CNN), COVID-19, X-Rays, Radiology**

I. Introduction

THE infectious disease COVID-19 is caused by the severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2). Since its initial identification in Wuhan, China in December 2019, it has been labelled as a pandemic by the World Health Organization. A high reproduction rate and a high mortality rate has resulted in government sanctioned border closures, deserted streets, panic-induced stockpiles, and a global recession.

Because the number of infected people is as an exponential function, it is critical to contain the virus through extensive testing. However, a scarcity and a mediocre true positive rate in standard tests, which involves a real-time reverse transcription polymerase chain reaction (rRT-PCR) from a nasopharyngeal swab, indicates the importance of alternate methods to diagnose the infection.

The paper explores the feasibility of a diagnosis of the infection through X-Ray imaging analysis of a lungs. The lungs are analyzed because the virus attacks epithelial cells in the respiratory tract. The analysis is done with Convolutional Neural Networks, a class of deep neural networks that are a regularized variety of multilayered perceptrons often used to analyze visual imagery. The architecture of multilayered perceptrons is similar to the neurons in the human brain and each neuron in a layer is connected to all neurons in the previous layer. However, the aforementioned characteristic of the multilayered perceptrons can result in an overfitted model. To combat the problem, Convolutional Neural Networks exploit the hierarchical nature of data to manage complexity with simpler structures. A convolutional neural network is used in the study because of its ability to extract features.

The performance of the convolutional neural network is evaluated with an independent data-set that is not involved in the development of the network to prevent bias. Furthermore, X-Ray images of patients with pneumonia not induced by COVID-19 are included to evaluate the robustness of the model. The performance of the model is measured with the F-1 score, sensitivity, and specificity, and the analysis is performed with Python libraries (Keras, TensorFlow).

Before the convolutional neural network is modeled, it is critical to realize that sensitivity, which is the measure of the proportion of positives that are correctly identified, is more important than specificity, which is the measure of the proportion of negatives that are correctly identified. While the consequences of a false positive are not that dangerous, the consequences of a false negative can result in deaths and increased infections.

Revised May 4, 2020

II. Prior Work

Multiple research teams have designed various convolutional neural network architectures to identify an efficient diagnosis method but, but most of the results are hidden, and, therefore, it may be difficult to evaluate their accuracy[2]. Methods to diagnose COVID-19 through X-Ray image analysis include:

A. Manual Analysis

A simple diagnosis method would be the manual analysis of X-Ray images to recognize COVID-19 infections. However, the limitation here is that the method has inferior accuracy and is time expensive. Furthermore, multiple radiology experts would be needed to analyse the images, which is not practical.

B. COVID-Net

There is a model called COVID-NET, which was developed by three researchers at the University of Waterloo in conjunction with a Canadian Artificial Intelligence company called DarwinAI[3]. They have decided to make their research and data-sets open-source and available to the public to accelerate the development of efficient diagnosis methods.

The architecture of COVID-NET is highly dependent on a design pattern known as projection-expansion-projection-extension (PEPX), which is able to achieve computational efficiency while maintaining high representation capabilities.

COVID-NET is trained on the COVIDx data-set, which consists of 13,800 X-Ray (CXR) images, created from three public data-sets. One problem here is a shortage of COVID-19 data. There are only 121 patients in the data-set with COVID-19, and only 183 COVID-19 CXR images in total. These are available for download at https://github.com/lindawangg/COVID-Net.

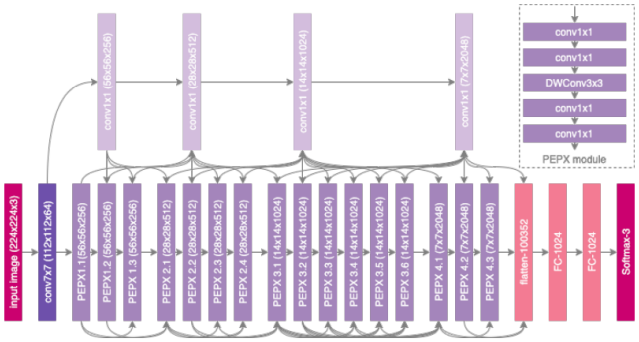

Fig. 1. COVID-Net Architecture

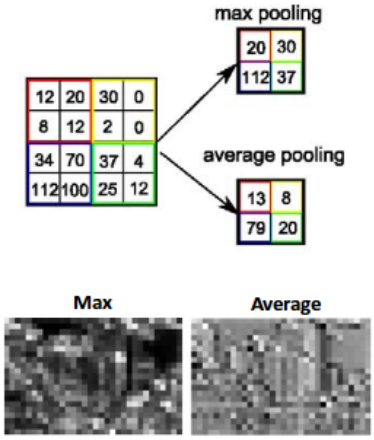

Fig. 2. Average Pooling Representation

C. VGG16

The Univeristy of Oxford's K. Simonyan and A. Zisserman developed the VGG16 in their paper "Very Deep Convolutional Networks for Large-Scale Image Recognition" that was submitted to the ImageNet Large Scale Visual Recognition Challenge 2014, a competition to classify images correctly. VGG16 analyses ImageNet, a data-set with 14,000,000 images and 1,000 categories, with 92.7 percent accuracy. VGG16 utilises several 3x3 filters.

III. Method

The paper uses a VGG16 convolutional neural network and incorporates a larger data-set, including data from Dr. Joseph Paul Cohen, a fellow at Université de Montréal. Datasets used are referenced [4][5].

Before the analysis, it was critical to decide that the analysis would be done on the Google Cloud Platform because training our VGG16 would be computationally intensive and it would be impractical to train and test our convolutional neural network on our local machine. VGG16 uses Keras and Tensorflow, which are common deep learning libraries that we installed on our cloud environment. Moreover, the matplotlib library is used to plot graphs and the OpenCV library is used to prepare the images in data-set.

The learning rate, the parameter that determines the step size in the algorithm that minimizes the loss, the number of epochs, the parameter that determines the number of complete forward and backward passes of the training set, and the batch size, the number of observations in a forward/backward pass, were initialized to be 0.001, 50, and 10 respectively.

After the images and categories that the images belong to were added to their respective arrays, the pixel intensities were normalized to the 0-1 range. Subsequently, the data is divided into a training set and a testing set. While 75 percent of the data is in the training set, 25 percent is in the testing set.

Furthermore, pooling layers are utilised. Pooling layers reduce the dimension of the data through the reduction of neuron clusters in the prior layer into a single neuron. Our model utilises average pooling, which uses the average of a neuron cluster in the prior layer in the current layer. This reduces over-fitting because irrelevant information is not considered.

After pooling, the pooled feature map is flattened, which involves the transformation of the pooled feature map into a single column.

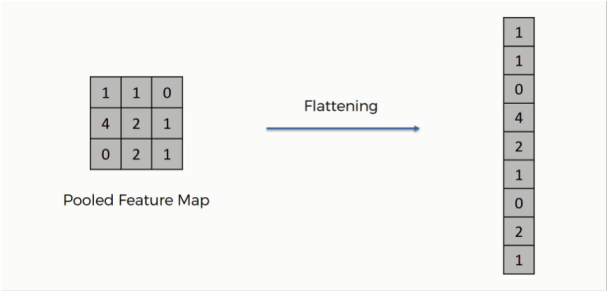

Fig. 3. Flattening Representation

After flattening, the single column is used as an input for the neural network. Herein, there is an input layer, fully connected layer, and the output layer. The output layer predicts the category to which the image belongs to. A rectified linear activation unit is used. The rectified linear activation unit benefits from its similarity to linear activation functions. Therefore, the optimization of the model is not difficult and the network nearly guarantees the prevention of gradient vanishing.

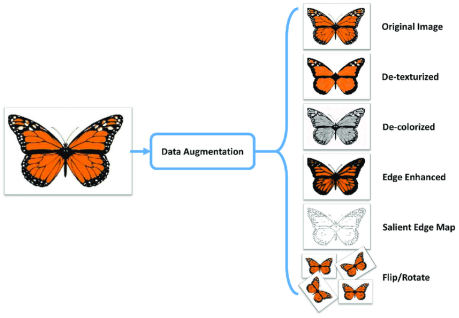

Fig. 4. Data Augmentation Representation

Since a binary classification is required, The prediction error is calculated via binary cross-entropy. Binary cross-entropy evaluates the deviation of the prediction from the true value

for each class. The deviation is averaged to evaluate the final loss. The error is backpropagated to improve the prediction. Backpropagation computes the gradient of the loss function and iterates backward to prevent gradient explosion or gradient vanishing.

The convolutional neural network is compiled with the Adam optimizer, which is time and memory efficient and uses adaptive estimates of lower order moments. Before the convolutional neural network is trained, data augmentation, which is the random transformation (shearing, rotation, scaling, and zooming) of data, is done to prevent over-fitting.

IV. Discussion And Analysis

This section highlights the numerical analysis conducted on the VGG16 Convolutional Neural Network with regards to the categorization of lung x-rays. The overall performance of the model turns out to be promising, considering the small dataset.

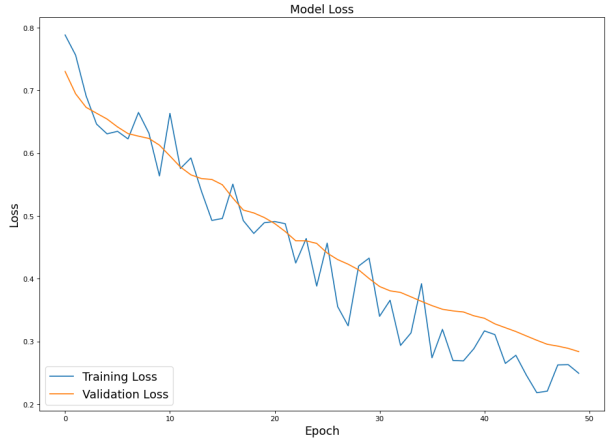

Fig. 5. Model Loss

Shown in Figure 5, is a plot of training loss (blue) alongside validation loss (yellow), versus time in epochs. This plot shows characteristics of a promising model for classification because training loss and validation loss both drop at a similar rate and approach close to 0—indication that the model is improving with each epoch. A slight divergence may be noted between training loss and validation loss near the 50th epoch, and while this may be a sign of slight over-fitting, it is acceptable because the validation loss continues to decrease with each epoch, indicting overall improvement of the classifier.

Figure 6 shows the training accuracy (blue) and validation accuracy (yellow) vs. time in epochs. Here it may be noted that the validation accuracy levels off around 90% around epoch 30, with only marginal accuracy improvement (if any) beyond this point. This indicates that the accuracy of the model levels off at slightly above 90%.

Figure 7 shows a classification report of the model. 100% precision, aka positive predictive value (PPV), indicates that the model was correct for every positive prediction it made (since PPV is the ratio of true positives to total positive predictions made). There were no false positives. This means

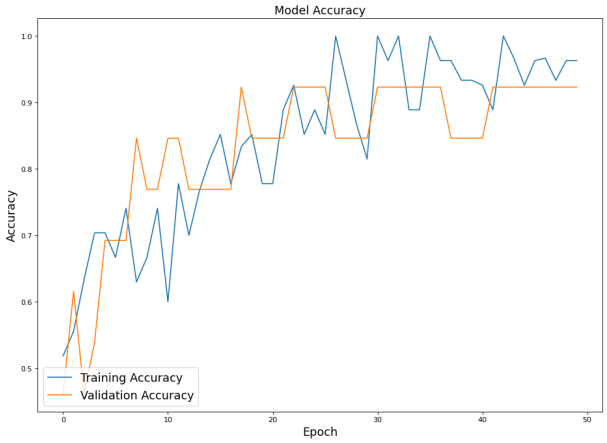

Fig. 6. Model Accuracy

that if this model makes a diagnosis that a patient is a carrier of COVID, this prediction would nearly always be correct.

While a PPV of 100% for COVID-19 classification is promising, recall, aka sensitivity, is a much more meaningful metric in the evaluation of this specific model. When it comes to medical testing, it's extremely important that the number of false negatives is close to 0—if a test result informs a patient that they're not infected with a virus when in reality they are, this may have devastating effects. The cost of a false positive diagnosis of COVID-19 could result in a patient being closely monitored, or forced to go through another test. A false negative diagnosis could cause a patient to believe they are free of the virus and spread it to others. Therefore, the cost of false positives in this specific medical situation is much lower than that of false negatives. The 83% recall metric indicates that only 83% of actual COVID-19 affected lungs were correctly labelled positive. 17% of cases with COVID-19 were incorrectly identified as 'normal', ie., they were false negatives. This value is troubling and indicates that this model is not suitable for production-level medical diagnosis.

	precision	recall	f1-score
covid	1.00	0.83	0.91
normal	0.88	1.00	0.93
weighted average	0.93	0.92	0.92
Accuracy	0.923		

Fig. 7.

Figure 7 also shows an accuracy of 92%. Accuracy is generally used as a performance measure when the number of true negatives and true positives is important, but here a better overall performance indicator is the F1 score. This metric seeks to provide a balance between precision and recall, and often acts as a point of reference when comparing two different architectures on the same dataset [4]. Due to time limitations, our team was not able to test other model architectures and provide any comparative analysis.

V. Limitations And Future Work

Although recent research has revealed that diagnosis of CT scans for COVID-19 can be more accurate than diagnosis of X-Ray images, the paper analyses the diagnosis of X-Ray images because X-Ray COVID-19 data-sets are more readily available. Furthermore, the authors were more equipped to analyse X-Ray images because of their experience in analysing X-Ray images previously. Also, the authors are not medical experts and are probably not equipped to decide on a reliable and practical diagnosis method.

Moreover, the COVID-19 X-Ray images data-set is limited, and, therefore, analysis of a more extensive data-set may reveal considerable alterations to our analysis. It should be noted that the true PPV is likely below 100%, but due to constraints on size of the dataset used, it is hard to determine the exact value.

Last, the project had to be done in limited time, and, therefore, other models and optimizers were not tested.

VI. Conclusion

In order to build an accurate COVID-19 classifer, we decided to employ a VGG-16 convolutional neural network architecture, based off it's reputation as one of the best vision model architectures. Previous work on COVID-19 detection has been done with a similar VGG-16 architecture, but we employed a larger dataset and managed to obtain marginally superior accuracy. Through our testing, we were able to conclude that it may be technically viable to classify COVID-19 induced respiratory damage with the use of a CNN model. Our model's main weakness was to identify normal lungs as COVID-19 ones. We had a false positive rate of 17% indicating that our model, with the scale of data that we employed, would never be feasible in a production environment. However, we used a relatively small training set, with less than 100 COVID-19 pneumonia samples, and it is not unreasonable to expect significant improvement with a substantially larger dataset. The initial performance of this model provides sufficient promise that further work with a larger dataset may result in a model which can accurately distinguish normal cases from COVID-19 cases.

Acknowledgment

The authors would like to thank their professor Fabien Scalzo and their teaching assistants Alexandre Tiard and Ali Atamizadeh. Furthermore, they would also like to thank the University of Oxford's K. Simonyan and A. Zisserman for the development of the VGG16 convolutional neural network.

References

[1] https://www.technologyreview.com/2020/03/12/905352/ai-could-help-with-the-next-pandemicbut-not-with-this-one/
[2] https://arxiv.org/pdf/2003.09871.pdf
[3] https://www.pyimagesearch.com/2020/03/16/detecting-covid-19-in-x-ray-images-with-keras-tensorflow-and-deep-learning/
[4] https://pathmind.com/wiki/accuracy-precision-recall-f1
[5] https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia
[6] https://github.com/ieee8023/covid-chestxray-dataset