Detection of COVID-19 with the Usage of Pre-trained CNN architectures

John Kim

***Abstract*— To prevent COVID-19 from spreading further, it is crucial to detect COVID-19 from patients as quickly as possible. Detection of COVID-19 from chest X-ray images can be an effective and efficient strategy to achieve this. Some of deep learning techniques including CNNs (convolutional neural networks) can be applied for detecting COVID-19 from chest X-ray images with training from chest X-ray images. However, with some careful attention, some of CNN models pre-trained on natural images may be able to be used to detect COVID-19 from chest X-ray images and to classify between COVID-19 positive and COVID-19 negative images. This article discusses a method for using some of pre-trained CNN models including Resnet-50, Resnet-101, Vgg-16, Vgg-19, Inception-v3, InceptionResnet-v2, and Xception to detect COVID-19 from X-ray images and how effective this method can be. Also, this study shows how k-fold cross-validation can affect the statistical measures of the classification. To summarize, as the major outcomes of this research, it was found that VGG-19 with k=3 had the highest sensitivity value (0.9667 = 96.67%), Resnet-50 with k=2 had the highest specificity value (0.9333 = 93.33%), and VGG-16 with k=7 had the highest accuracy value (0.9148 = 91.48%).**

***Keywords* - CNNs, K-fold cross-validation, COVID-19, Chest X-ray images**

John Kim is a third year electrical engineering student at the University of British Columbia, BC, Canada (e-mail: johnkm37@gmail.com).

I. Introduction

To diagnose COVID-19 from an individual, PCR, molecular polymerase chain reaction test, is predominantly employed. It takes 1 to 3 days to acquire the results from this test [1]. However, an individual with COVID-19 can be contagious 2 day to 3 days before having symptoms [2]. To prevent any potential contagion, it would be beneficial to quickly determine if an individual is COVID-19 positive and if he or she needs to be quarantined as soon as possible. One helpful approach to this would be to use X-ray chest images to detect COVID-19 for prognosis purpose. On average, it takes fifteen minutes for one single X-ray examination process including positioning a patient, obtaining and verifying the images [3]. Before carefully examining this method, it is important to acknowledge that X-ray chest images may not be able to completely diagnose COVID-19 because consolidation, one important feature for this method, can be caused by other diseases including bacterial or other viral pneumonias. However, still, for cases where PCR tests are not available, it may be beneficial to use X-ray images to determine whether a patient has COVID-19 or not. Based on the current literature, one team in Turkey achieved up to 99.7% with 5 fold validation and Resnet-50 for classifying between COVID-19, normal (healthy), viral and bacterial pneumonias [11]. Another team from the USA and Iran applied multiple CNNs, transfer learning and achieved a sensitivity rate of 98 plus minus 3% and a specificity rate of 90 percent [12].

II. Summary of Project Work and Contributions

This project employs multiple CNN architectures to extract features from COVID-19 positive AP X-ray images, COVID-19 positive non-AP X-ray images, and COVID-19 negative AP X-ray images. With these features, a multiclass SVM (support vector machine) is trained and used for making predictions. The results of this process are analyzed in terms of sensitivity, specificity, and accuracy. With these statistical measures, it is possible for one to compare and contrast the performances of CNN architectures classifying each class of images.

III. Materials

A. Chest X-ray images with COVID-19 Positive

All the COVID-19 positive chest radiographs were acquired from a GitHub database run by Dr. Cohen, postdoctoral fellow from University of Montreal [4]. His database had different types of chest images. Four types of chest images were obtained and employed for this project. They were anteroposterior (AP), posteroanterior (PA), lateral X-ray images and CT-scans. 100 images for each case were acquired. The main X-ray image type used for this project was AP type. Pneumonia-infected lungs display a white spot in AP chest X-ray images. This white spot refers to a consolidation. In fig1, this consolidation is indicated by arrows. As examples, the other types of chest images are displayed in Fig2, Fig3, and Fig4.

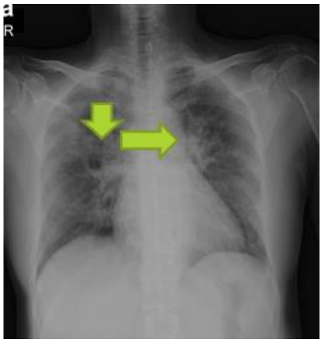

Fig.1 COVID-19 positive AP chest X-ray image with consolidation indicated by the arrows.

Fig.2 COVID-19 negative AP chest X-ray image. There is no consolidation or white spot in this image.

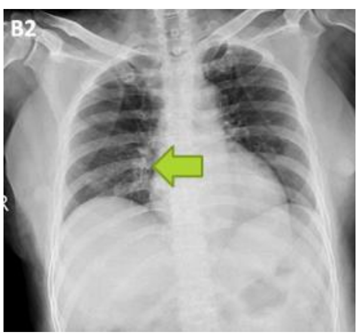

Fig.3 COVID-19 positive PA chest X-ray image. A consolidation spot is indicated by the arrow.

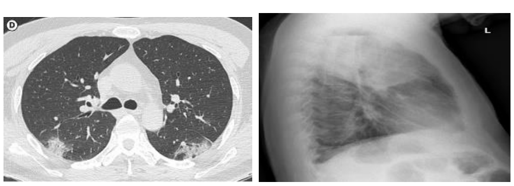

Fig.4 COVID-19 positive CT and lateral chest X-ray images.

B. Chest X-ray images with health lungs

100 X-ray images with healthy lungs were acquired from a Kaggle dataset posted by Paul Mooney, developer advocate at Kaggle [5]. They were all AP X-ray images. An example is shown in Fig 2.

IV. Methods

A. Deep-learning and Detection of COVID-19

Dr. Barath Narayanan, a research scientist at UDRI's software systems group at University of Dayton, suggests a method for classifying between COVID-19 infected lungs and non-COVID-19 from chest X-ray images by applying an available CNN (Convolutional Neural Network) architecture, Resnet-50 in MATLAB [6]. According to his work, he uses k-fold cross-validation to split his dataset for a quality estimate of performance of Resnet-50. He provides resnet-50 with this dataset to train and test this CNN architecture. He employs transfer learning with Resnet-50. Based on the results of this process, he examines the performance of this architecture.

B. K-fold cross-validation

To clarify some of the steps of Dr. Barath Narayanan's method, k-fold cross-validation is a statistical method for resampling one's data to have less bias for testing and training his or her models. With k-fold cross-validation, one can split one's data into blocks and can use all the blocks for testing purpose. This allows him or her not to be concerned about which block of the data would be best for testing purpose. For instance, in Fig.5, there are five different blocks of data [7]. One can use each individual block to test and the rest for training. After he or she completes the testing process with every block, this individual can summarize the overall results by combining the result from each block with one another.

C. Resnet-50

The CNN architecture used for Dr. Barath Narayanan's research, Resnet-50, is a CNN architecture that can effectively treat vanishing gradient problem. As the depth of neural networks increases, it becomes more challenging to train due to vanishing gradient problem. As the gradient gets backpropagated to earlier layers, iterated multiplication can make the gradient very small. This can make network performance saturated and degrade. To resolve this, Resnet-50 applies skip connection method. This method stacks convolutional layers together and adds a shortcut from the original input to the output of the convolution block and combines the original input with the output of the convolution block. By doing so, this process can reduce vanishing gradient by giving the gradient this shortcut for going through. This shortcut can be seen in Fig 6 [8].

D. Algorithm of this Project

Some portions of this project were inspired by Dr. Barath's work. For this project, 7 different CNN architectures were used for extracting the features of each category of images. Also, transfer learning was not applied. For the training part, a multiclass SVM (support vector machine) was used. SVMs are supervised machine learning algorithms that can be applied for classification problems [9]. As the first step of this project, all the sets of COVID-19 positive and negative X-ray images were loaded. With different k values, these sets were divided through k-fold cross validation. All the CNN architectures were provided with these divided datasets. In order to examine their performances, confusion matrices were computed. From the matrices, true positive (TP), true negative (TN), false positive (FP), and false negative (FN) were extracted. Based on these data, accuracy, specificity, and sensitivity were computed and used for comparing the performances of CNN architectures and to examine how different k-values affected the results.

V. Implementation Details

A. Loading Images

Every code for this project is written in MATLAB. In order to load the image datasets, imageDatastore function was used. This function builds a datastore from collecting image datasets. A datastore is a repository used for storing, managing sets of data. There were three different subdirectories for this image datastore, one for AP X-ray images with COVID, another for AP X-ray images without COVID, and one for non AP X-ray images with COVID.

B. K-fold cross validation

The implementation of k-fold cross-validation were inspired by Dr. Barath Narayanan's work [6]. Based on his work, he set the k value first and indices for all the images and test images first. Then, he excluded all the test indices from the entire image indices. This leaves one with only the indexes for training set. This process repeats with different sets of indices for the testing set and training set and results in a k-fold cross validation. In this project, a while loop was built in the code in order to implement this idea with different testing set indices. To change the k-value, a for loop was implemented with the while loop nested inside.

C. Extraction of Features, Training, and Prediction

Some portions of this algorithm were learned from the MATHWORKS help center website [10]. With the datasets from the cross validation, 7 different CNN architectures were used for this project, namely, Resnet-50, Resnet-101, Vgg-16, Vgg-19, Inception-v3, InceptionResnet-v2, and Xception. They are all available in MATLAB deep learning toolbox. The architectures were applied for extracting specific features from the different training sets. To achieve this, activations function in MATLAB was applied. This function calculates deep learning network layer activations. A specific layer had to be set for this function. The layer right before the classification layer was chosen to have full numbers of layers for every architecture. Fitcecoc function in MATLAB can train a multiclass SVM with training data sets. With the training images from the cross validation process, a multiclass SVM was trained. To test this SVM, predict function was used. It was provided with the SVM and the extracted training features.

D. Confusion Matrices and Measurements

Before making confusion matrices, all the results from the testing process were collected from each k-fold cross validation. Confusion matrices were computed for different k-values. From the matrices, all the TP, TN, FP,FN were extracted. With these values, accuracies, sensitivities, and specificities were compute by the equations from Fig 7.

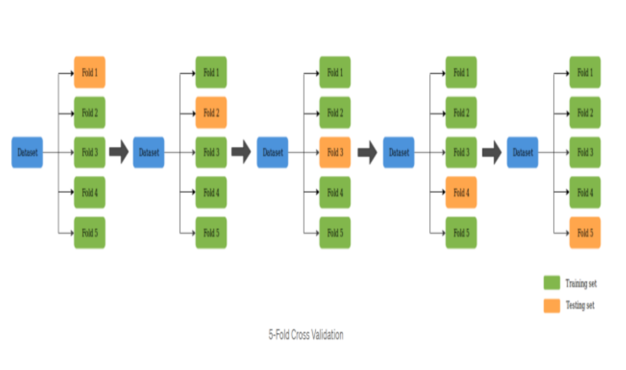

Fig.5 k-fold example with 5 folds.

Source: Adapted from [7]

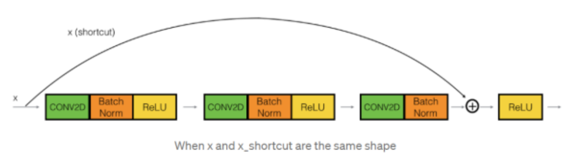

Fig.6 Resnet-50 shortcut shown above.

Source: Adapted from [8]

(1). Sensitivity = True Positives/(True Positives + False Negatives)
(2). Specificity = True Negatives/(True Negatives + False Positives)
(3). Accuracy = (True Positives + True Negatives)/(True Positives + True Negatives + True Negatives + False Positives)

Fig.7 Sensitivity, specificity, and accuracy equations.

VI. Results

A. Confusion Matrices

As suggested in class, the informal rule of thumb for the ratio of training and testing sets (80 and 20 percent) was initially used for the result. Therefore, k was set to be 5 because we wanted to achieve 80 per cent of training and 20 per cent of testing sets.

Fig 8 to Fig 14 show the confusion matrices of ResNet50, Resenet101, Vgg16, Vgg19, inceptionv3, inceptionresnetv2, xception correspondingly. (discussion continues in page 7)

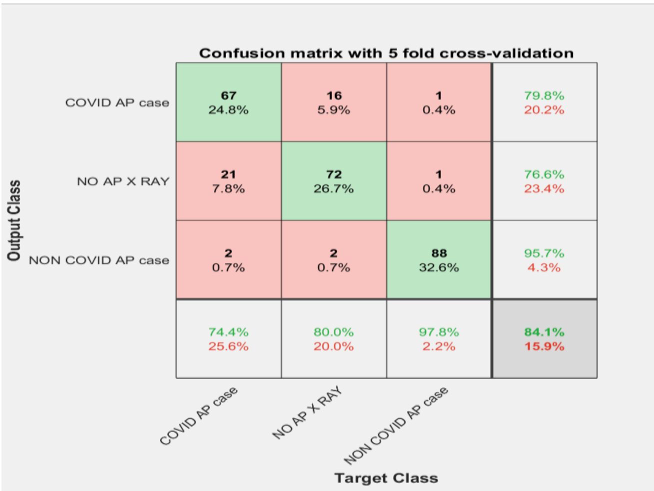

Fig 8. Confusion matrix of Resnet-50 with k=5.

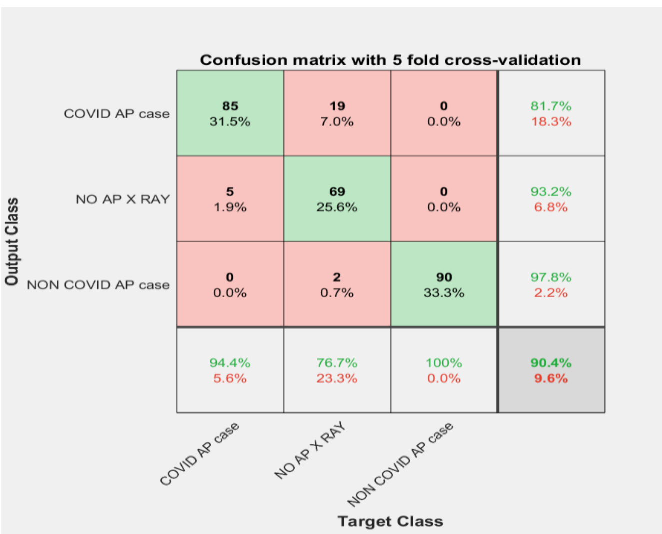

Fig 9. Confusion matrix of Resnet-101 with k=5.

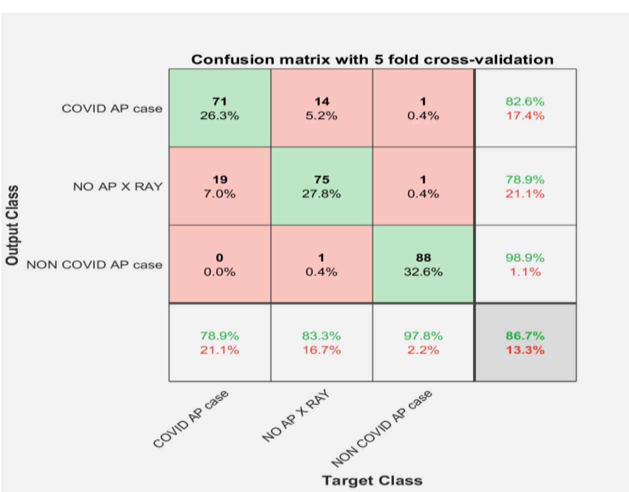

Fig 10. Confusion matrix of VGG-16 with k=5.

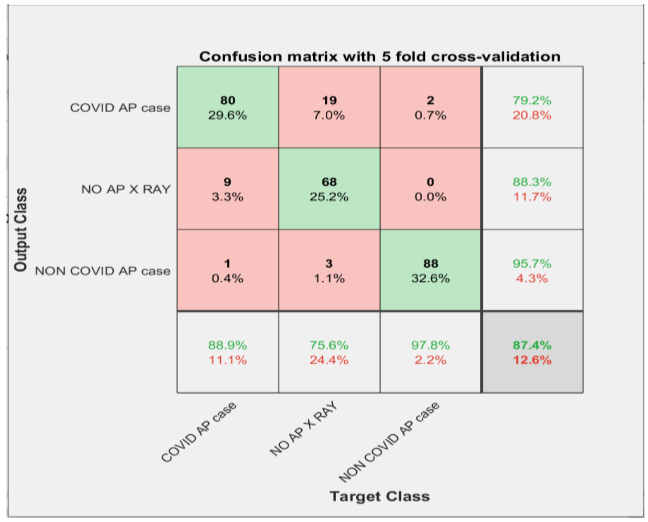

Fig 11. Confusion matrix of VGG-19 with k=5.

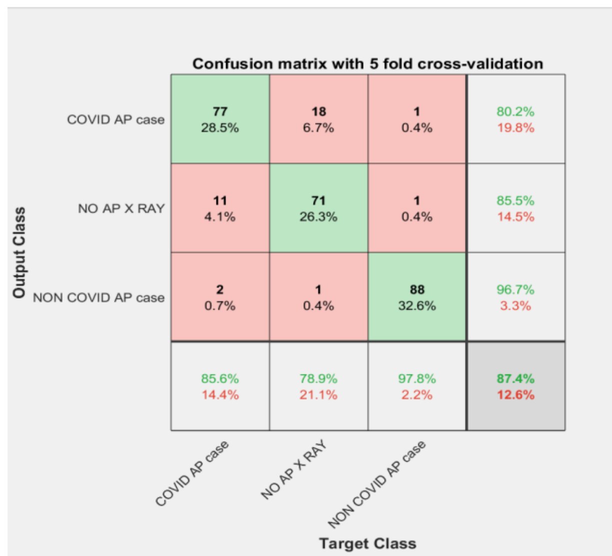

Fig 12. Confusion matrix of Inception-v3 with k=5.

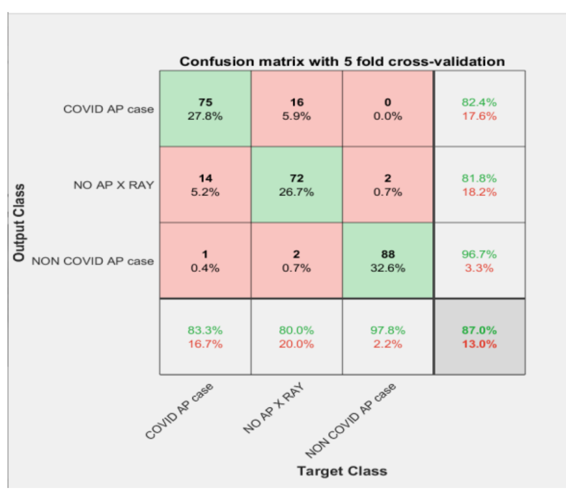

Fig 13. Confusion matrix of Inception Resnet v2 with k=5.

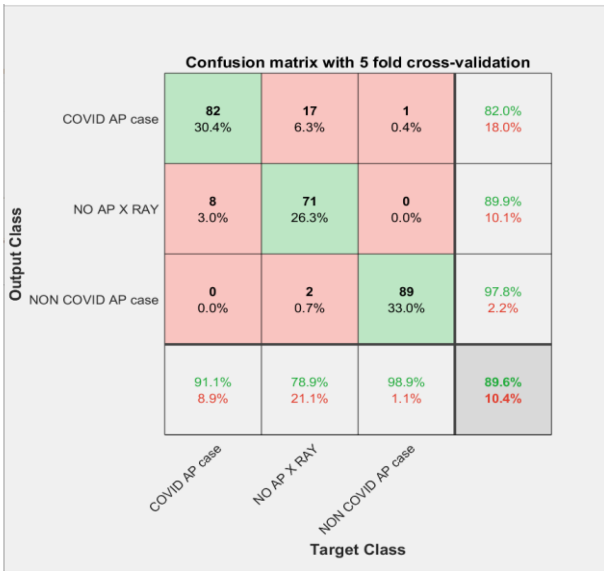

Fig. 14 Confusion matrix of Xception with k=5.

(Continued discussion) No AP X-ray represents non-AP chest X-ray images with COVID-19. The y-axis of this matrix indicates the predictions made. The x-axis is responsible for the actual data. Based on the matrices, most of the non-COVID-AP class were correctly classified although there were 1 to 3 faulty classification cases. For instance, from the confusion matrix of Xception, one out of 90 cases of non-COVID AP class were misclassified. However, some of the COVID-AP class and non AP class images were misclassified as one another. For example, from fig 13, 16 No-AP cases were misclassified as AP cases while 14 COVID-AP cases were misclassified as No-AP cases. This was potentially caused by a similarity between AP and PA X-ray images. As possibly suggested in fig 2 and fig 4, an AP and a PA chest X-ray images resemble one another in their shapes. Therefore, this resemblance could make it potentially challenging for the CNNs to differentiate their features from one another. To confirm this, Resnet-50 was used to show if it could classify between COVID-19 AP and PA cases.

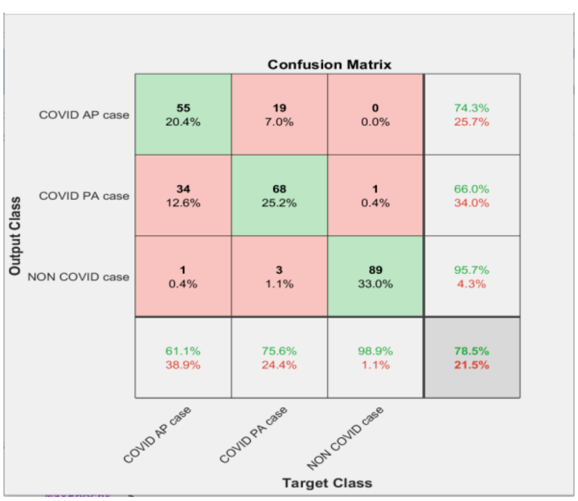

Fig. 15 Confusion matrix of Resnet-50 with COVID PA and COVID-AP classes.

The results are shown in fig 15. Based on the result, approximately a quarter to one third of either COVID-19 AP or COVID 19 PA class images were misclassified as one another. This finding potentially suggests that some of the CNN models have a limitation when it comes to dealing with more field-specific details. In this case, the field specific details would be restricted to radiology field. The reason why this potential limitation arose was possibly because the CNNs used for this project were pre-trained based on natural images. It could have been more effective if they were trained based on more field specific data such as X-rays in this case. Another potential reason was because more layers were not explored when the activations function was applied. The layer before the classification layer might not have an ideal decision to make for that function. Regardless of this limitation, more potentially meaningful data were calculated.

B. Sensitivities, specificities, and Accuracies

Sensitivity in this context shows what percent of AP images with COVID 19 were correctly identified. Specificity indicates what percent of other cases except for AP images with COVID 19 were correctly identified. Lastly, accuracy shows us the percent of every class being correctly identified. Figs 16 to 18 show their plots. Based on the plots, Resnet-101 has the highest sensitivity and accuracy values while VGG-16 has the highest specificity value. To see if different k values from the k-fold cross-validation process resulted in different ranks of sensitivity, specificity, and accuracy, sensitivities, accuracies, and specificities were found with different k values from 2 to 10. In other words, the ratio of testing and training sets were changed from 50:50 to 10:90 (testing : training). Since there were too many plots for this, it was decided to show only the plots with major results. Based on figs 19 and 20, Resnet-101 does not have the highest sensitivity value across all the k-values. This trend repeats in specificity and accuracy although the plots could not be shown because of space. The previous models with the highest accuracy and specificity did not have the highest values across all the k-values. According to the results, one observation could be made. It can be challenging to find the optimal k-values for estimating the best statistical measures of CNNs. This could be possibly because k-fold cross-validation resulted in less bias by testing different data sets, resulting in not having biased statistical measures. However, to clarify, across all the k-fold values, VGG-19 with k=3 had the highest sensitivity value (0.9667 = 96.67%), Resnet-50 with k=2 had the highest specificity value (0.9333 = 93.33%), and VGG-16 with k=7 had the highest accuracy value (0.9148 = 91.48%).

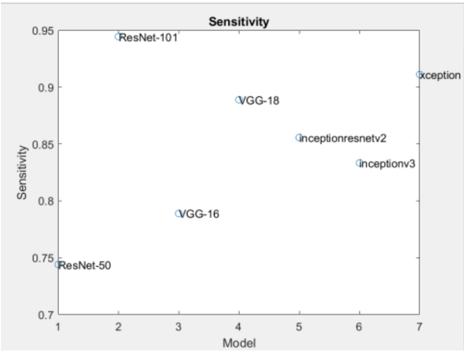

Fig. 16 Sensitivities vs CNN Model.

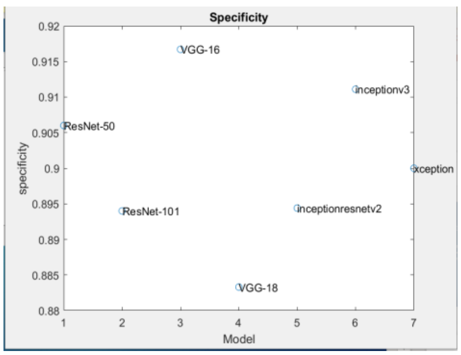

Fig. 17 Specificity vs CNN Model.

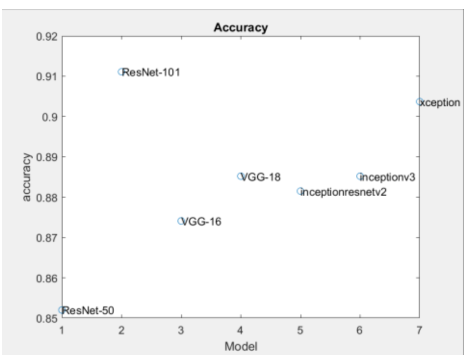

Fig. 18 Accuracy vs CNN Model.

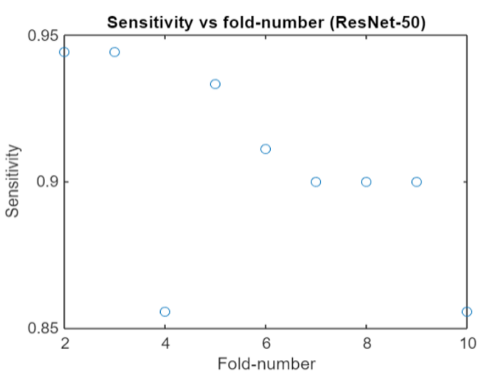

Fig. 19 Sensitivity vs Fold-number (Resnet-50).

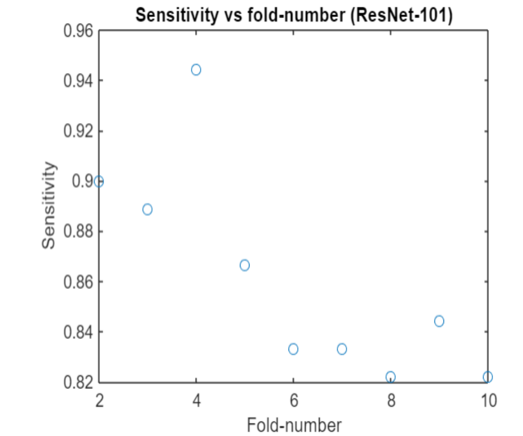

Fig. 20 Sensitivity vs Fold-number (Resnet-101).

VII. Conclusion

Through this project, seven pre-trained CNN models were employed to extract features from COVID-19 positive AP X-ray images, COVID-19 negative AP X-ray images, and COVID-19 positive non-AP X-ray images. These features were used to classify the different classes of images.

Based on the results, it was observed that VGG-19 with k=3 with the highest sensitivity value (0.9667 = 96.67%), Resnet-50 with k=2 with the highest specificity value (0.9333 = 93.33%), and VGG-16 with k=7 with the highest accuracy value (0.9148 = 91.48%). However, it was noticeable that the changes in k-values resulted in different orders of magnitude for the statistical measures. One potential limitation was also found. The CNN models pretrained on natural images may be able to spot conspicuous features such as consolidations. However, they may not be very helpful for any specific field-related details such as the PA vs AP difference. If a chance is given to further this project, it would be a good idea to build a CNN model from scratch and to train it with chest X-ray images only. Simultaneously, it would be another good option to use transfer learning and fine-tuning to improve the quality of results.

Acknowledgment

I would like to deeply appreciate Dr. Garbi’s insightful advice and patience. Whenever I asked her for tips and advice for this project and ELEC 421, without a hesitation, she responded with kindness and wisdom.

REFERENCES

[1] Testing for COVID-19: Diagnosing and how we test, Government of Canada, [online], 2020,

Available: https://www.canada.ca/en/public-health/services/diseases/2019-novel-coronavirus-infection/symptoms/testing/diagnosing.html

[Accessed Dec.10,2020].

[2] If you've been exposed, are sick, or are caring for someone with COVID-19, [online], 2020,

Available: https://www.health.harvard.edu/diseases-and-conditions/if-youve-been-exposed-to-the-coronavirus#:~:text=A%20person%20with%20COVID%2D19,behaviors%20designed%20to%20prevent%20spread

[Accessed Dec.10,2020]

[3] X-ray (Radiography) – Chest , [online], 2020,

Available:https://www.radiologyinfo.org/en/info.cfm?pg=chestrad#:~:text=The%20entire%20chest%20x%2Dray,any%20changes%20in%20the%20chest

[Accessed Dec.10,2020]

[4] COVID-19 image data collection, [online] 2020,

Available: https://github.com/ieee8023/covid-chestxray-dataset

[Accessed Nov,2020]

[5] Chest X-ray Images (Pneumonia), [online], 2017,

Available: https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia

[Accessed Dec.10,2020]

[6] Deep Learning for Medical Imaging: COVID-19 Detection, [online], 2020,

Available: https://blogs.mathworks.com/deep-learning/2020/03/18/deep-learning-for-medical-imaging-covid-19-detection/

[Accessed Nov,2020]

[7] K-Fold Cross Validation, [online], 2018,

Available: https://medium.com/datadriveninvestor/k-fold-cross-validation-6b8518070833

[Accessed Dec.10,2020]

[8] Understanding and Coding a ResNet in Keras, [online], 2019,

Available: https://towardsdatascience.com/understanding-and-coding-a-resnet-in-keras-446d7ff84d33

[Accessed Dec.10,2020]

[9] What is a Support Vector Machine, and Why Would I Use it?, [online] ,2017,

Available: https://www.kdnuggets.com/2017/02/yhat-support-vector-machine.html

[Accessed Dec.10,2020]

[10] Image Category Classification Using Deep Learning, [online]

Available: https://www.mathworks.com/help/vision/ug/image-category-classification-using-deep-learning.html

[Accessed Nov,2020]

[11] Ali Narin, Ceren Kaya, and Ziynet Pamuk, "Automatic Detection of Coronavirus Disease (COVID-19) Using X-ray Images and Deep Convolutional Neural Networks" ,2020. Accessed on: Dec 5,2020. [online]. Available: https://arxiv.org/abs/2003.10849

[12] Shervin Minaee, Rahele Kafieh, Milan Sonka, Shakib Yazdani, and Ghazaleh Jamalipour Soufi, "Deep-COVID: Predicting COVID-19 from chest X-ray images using deep transfer learning", 2020. Accessed on: Dec 5,2020. [online]. Available: doi: 10.1016/j.media.2020.101794