COVID-19 Detection Using Chest X-Ray Images and CNN

國立政治大學 應用數學系 林煜翰

Jan 18, 2020

Abstract

Emerging from China, the novel coronavirus (COVID-19) has spread rapidly around the globe. The total confirmed cases worldwide, according to the World Health Organization (WHO), has exceeded 9 million and the death toll is approaching 500,000. Although governments around the world has been taking extreme effort to prevent the virus form spreading even more, it seems that the coronavirus case rate is not slowing down any time soon. One huge obstacle we are facing now is the shortage of COVID-19 testing kits. Moreover, it takes days, even weeks, to receive the results. Thus, it will be extremely convenient to implement an automatic system to quickly diagnosis the virus. In this project, five per-trained neural network models (VGG16, ResNet50, InceptionV3, DenseNet121, CheXNet) have been chosen for the detection of COVID-19 patient using their frontal chest X-ray radiographs. Since the datasets are limited, transfer learning is applied. Confusion matrix and 10-fold cross validation are proposed for the analyses of the models. The result is surprising. VGG16 triumph among the five neural network models with a 99% F1-score, the highest classification performance.

1 Introduction

The novel coronavirus at first was called severe acute respiratory syndrome coronavirus 2 or SARS-CoV-2 and was later renamed as COVID-19. Coronaviruses are a huge family of viruses that can cause respiratory tract infections. The illness including, mildly, some cases of common cold and, severely, SARS, MERS and COVID-19. Signs of infection include respiratory symptoms, fever, cough and dyspnea. In more severe cases, the infection can cause pneumonia, severe acute respiratory syndrome, multi-organ failure and death. Though up to 82 percent of the cases in COVID-19 are mild symptoms, the others are server or critical. Today COVID-19 cases total at 9,129,146 worldwide and the deaths approximate at 473.797.

Diagnoses, as for now, include everyone whose chest scan reveals pneumonia pattern instead of just the positivity of the virus tests. Tough some COVID-19 patients avoid death, they are left with a permanent lung damage. As the World Health Organization (WHO) reported, COVID-19 patients' lungs are found to be in a state of a "honeycomb-like appearance" similar to the SARS patients' lungs.

In the experiment done by Souradip Chakraborty on Medium, he choose VGG16 as his neural network which it's classification between normal chest X-ray and covid-19 chest X-ray yields a 100% accuracy and F1-score and it's classification between pneumonia chest X-ray and covid-19 chest X-ray yields a 91% accuracy and F1-score. However, the experiment only uses in total 276 images, with 100 normal, 100 pneumonia and 76 COVID-19 patients chest X-ray(?, ?) . Another study done by Zonguldak Bulent Ecevit University compared the three models, including ResNet50, InceptionV3 and Inception-ResNetV2. They found out ResNet50 results in a highest performance with a 97% accuracy, whereas InceptionV3 and Inception-ResNetV2 yields 97% and 87% respectively. Again, in this study they only choose 50 images each classes with a total of 150 data(?, ?) . The two attempts of COVID-19 detection using neural network mentioned above is what inspired this project. The goal to this experiment is to test which model structure in this case is more capable and to create a robust and accurate model for doctors to apply a fast automatic classification process to identify COVID-19 patients.

2 Datasets

The frontal chest X-ray data contain three classes:

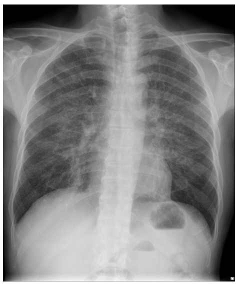
1. Frontal chest X-ray of normal healthy people
2. Frontal chest X-ray of patents with general pneumonia
3. Frontal chest X-ray of confirmed COVID-19 patents

Which are labeled as normal, pneumonia and covid-19 respectively.

The normal and pneumonia frontal chest X-ray data were selected from Kaggle repository "Chest X-Ray Images (Pneumonia)" (?, ?)and the covid-19 images were obtained from the open source GitHub repository shared by Dr. Joseph Cohen(?, ?). This repository consists chest X-ray / CT images of patients with acute respiratory distress syndrome (ARDS), COVID-19, Middle East respiratory syndrome (MERS), pneumonia, severe acute respiratory syndrome (SARS). The dataset contains a total of 600 training data and 90 testing data (200 and 30 images each case respectively). All images are resized into size 224x224. Figure 1(a), Figure 1(b) and Figure 1(c) showcase the normal, pneumonia and covid-19 frontal chest X-ray.

(a)

(b)



(c)

Figure 1: (a) normal (b) pneumonia (c) covid-19

3 Transfer Learning

Transfer learning is the technique to retain n-layers of weights in a pre-trained model and apply such model on a different set of data. The concept derives form the observation of the phenomenon that the modern deep neural network the features of the trained first-layer tend to resemble either Gabor filters or color blobs. This occurs not only for different datasets but the different objectives. And in studies it is proven that the transferring a pertained features form each layer of a neural network and then apply to different data yields better performance.

The models utilized in this experiment for transfer learning are VGG16, ResNet50, InceptionV4, DenseNet121 and CheXNet. Except CheXNet, the others are all pre-trained with ImageNet. CheXNet is created by the Standford ML Groups, which the structure was base off of DenseNet via transfer learning with 112,120 frontal-view X-ray images containing 30,805 unique patients and 14 classes. DenseNet is a neural network containing "Dense Block". A "Dense Block" is can be described as a more advanced "Residual Block" form ResNet. Instead of taking only the last layers output as input, "Dense Block" takes all the preceding feature-maps input.

The reason CheXNet was chosen is mainly because the database it pre-trained with is the exact same type of the data we are using here. Logically it should perform the best.

4 Experiment

All the programming is written in Python and executed in Google Colaboratory using the given GPU. With the exception of CheXNet, all the other pre-trained models are obtained in Keras application. The pre-trained CheXNet model is acquired from an open source Github repository(?, ?). The input of the images were equally set to be 224x224. To train the model we, first, frozen all the weights before the flattening layer, here we changed it into the Global Average Pooling 2D layer, and only train the two added fully connected layers and the output layer using Adam optimizer displayed in Figure 2. The batch size, learning rate and number of epochs were experimentally set to 15, 0.01 and 30, respectively for all experiments with an early-stopping set as validation accuracy without updating in 5 epochs.

10-fold cross validation was performed to ensure the robustness of the models. In which the dataset was spited randomly into 10 sections. Each section was set as the testing in cross validation in each fold. The result of each folds' loss and accuracy were averaged in the end as shown in Figure 3(?, ?).

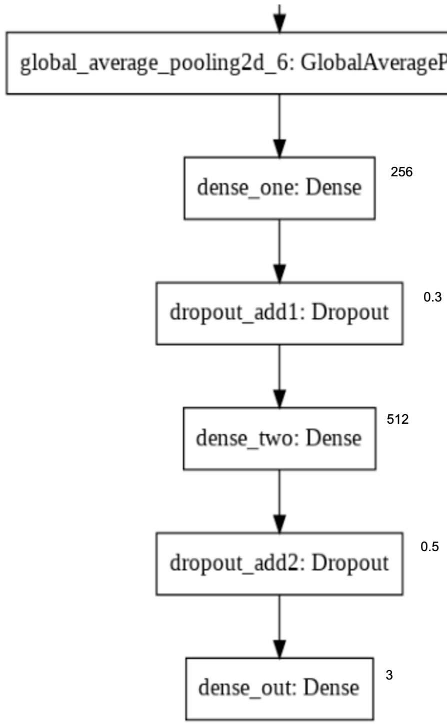


Figure 2: The add layers with the following order after the Global Average Pooling layer : Dense 256, Dropout 0.3, Dense 512. Dropout 0.5 and the output layer with three classes

Figure 3: 10-fold cross validation with the accuracy averaged in the end

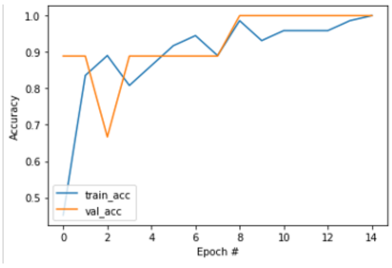
5 Results

5.1 Cross Validation

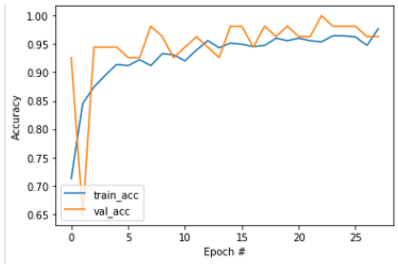
Five neural network models were compared in this study, including VGG16, ResNet50, InceptionV3, DenseNet121 and CheXNet. Training accuracy and loss value for the 5th fold of the pre-trained models are given in Figure 4 and Figure 5 respectively. From the graph we can see that all of the models in training are robust and not over fitting. From Table 1 we observe that VGG16 yields the highest test accuracy mean and what's interesting is that VGG16s Train accuracy mean is lower than Test accuracy mean. The weights in the models are updated by cycling through the epochs, which means the weights are changing according to the training data. That is, the accuracy should be higher with the training data. However, in this case it is the opposite. There are a few reasons that might result in such phenomenon. First, the dropout in Keras' predict is disable by default, since the propose of the dropout layer is to add more divergence in the models structure to fine tune the model. Therefore, usually when fitting the testing data the dropout will be disabled. Now, because the data here is unique and limited the dropout rate set in this case is high (dropout rate = 0.5). Thus, the this is the most likely reason that cause such result. Second reason is the data separation is uneven. It is unlikely this is the reason in this case since the experiment here has a 9 to 1 ratio which is very common. Last but not least, it may just be because the test data is simply easier to classify. The last reason is also unlikely since we've trained a few other models without this happening.

	Train accuracy mean	Validation accuracy mean	Test accuracy mean
VGG16	0.8847831004858018	0.8414814899365108	0.9766666650772095
ResNet50	0.9105317454482383	0.9214945067237693	0.8599999904632568
InceptionV4	0.904927206163307	0.899257408685855	0.7177777767181397
DenseNet121	0.9243752146834693	0.9162197955229081	0.7300000071525574
CheXNet	0.7675613627518102	0.8289563370278323	0.7555555582046509

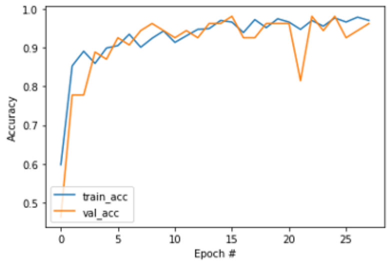
Table 1: The mean of the results of the 10-fold cross-validation



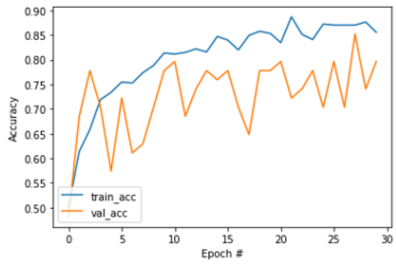
(a)



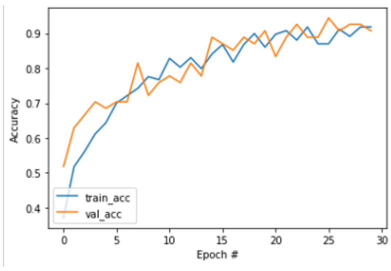
(b)



(c)

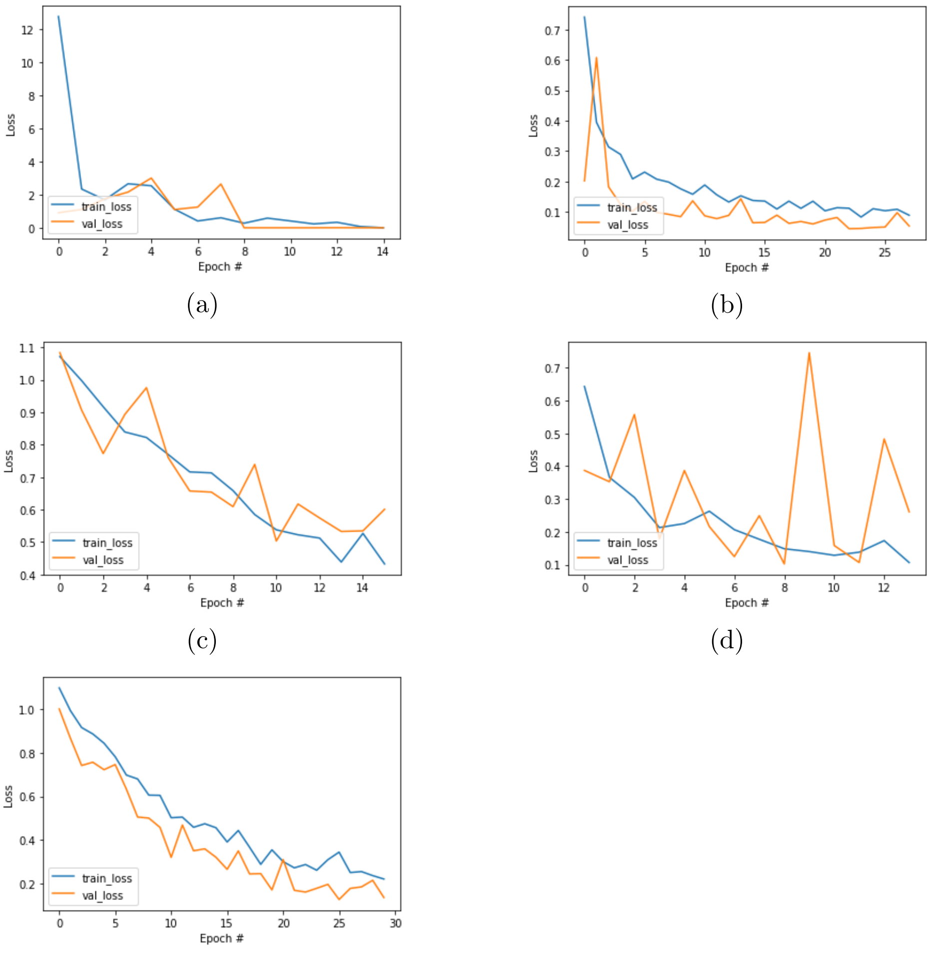


(d)



(e)

Figure 4: The 5th fold training and validation Accuracy of the five pre-trained models:
(a) VGG16 (b) ResNet50 (c) InceptionV3 (d) DenseNet121 (e) CheXNet



(a) (b) (c) (d) (e)

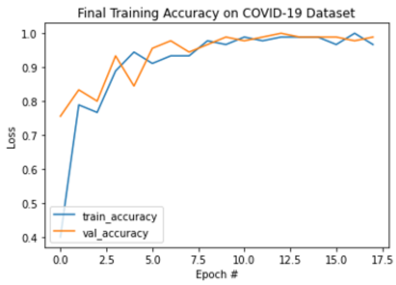
Figure 5: The 5th fold training and validation Loss of the five pre-trained models:
(a) VGG16 (b) ResNet50 (c) InceptionV3 (d) DenseNet121 (e) CheXNet

5.2 Full Dataset Training

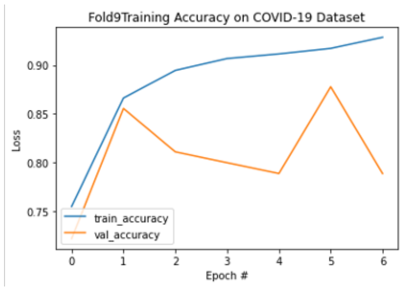
The reasoning for performing both cross validation and full dataset training is because the purpose of this project is to create a robust and accurate model to be easily utilized. The cross validation set is mostly for the robustness and the full dataset is for actually building the model for usage. In here we can observe that surprisingly VGG16, the simplest structure among all the other models yields the best result with a whopping 99% accuracy and F1-score. Since this is an extremely special case, with a limited data only 3 classifications and build on top of transfer learning, this could be the reason. The neural network are mainly trial and error and data driven, it is hard to tell whether it is the models structure problem or data itself. However, this result has achieved what our goal requires which is finding an accurate and robust model, therefore, although it is kind of a hit by mistake, it is mission complete.

	Precision	Recall	Accuracy	F1-score
VGG	0.99	0.99	0.99	0.99
ResNet50	0.83	0.79	0.79	0.79
InceptionV4	0.76	0.76	0.76	0.74
DenseNet121	0.81	0.80	0.80	0.79
CheXNet	0.84	0.81	0.81	0.80

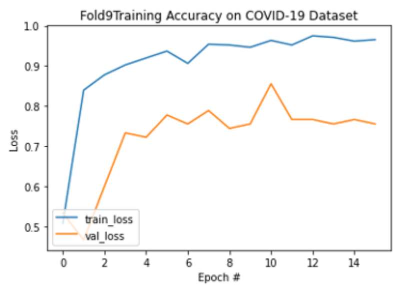
Table 2: The result of the full dataset transfer learning



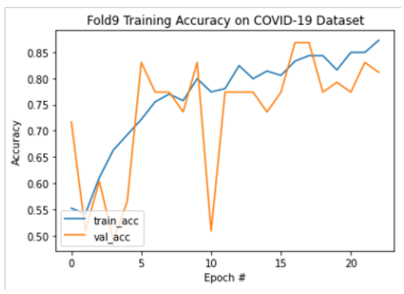
(a)



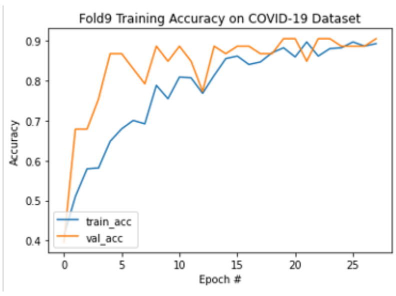
(b)



(c)

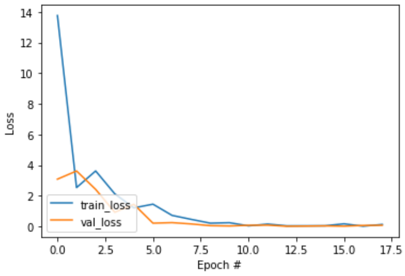


(d)

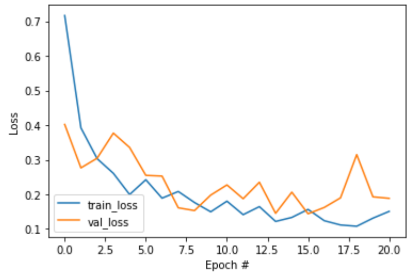


(e)

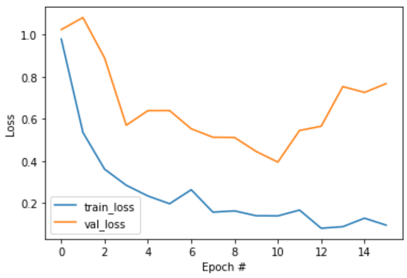
Figure 6: The full dataset training and validation of the five pre-trained models:
(a) VGG16 (b) ResNet50 (c) InceptionV3 (d) DenseNet121 (e) CheXNet



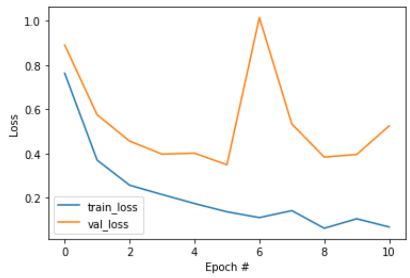
(a)



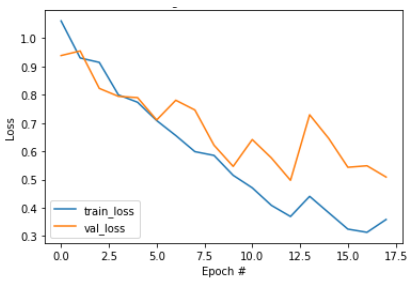
(b)



(c)



(d)



(e)

Figure 7: The full dataset training and validation Loss of the five pre-trained models:
(a) VGG16 (b) ResNet50 (c) InceptionV3 (d) DenseNet121 (e) CheXNet

6 Conclusion

Fast automatic diagnose of COVID-19 is vital for preventing the spread of the virus. This experiment compared five different CNN models VGG16, ResNet50, InceptionV3, DenseNet121 and CheXNet using transfer learning on classifying three classes of frontal chest X-ray images including COVID-19 patients X-ray, normal people X-ray and general pneumonia patients' X-ray. We observed that the VGG16 pre-trained model using imagine weights results in the highest performance with a 99% F1-score and accuracy. The pre-trained CheXNet trailing behind with a 80% and 81% of F1-score and accuracy respectively. And other models average at 78% accuracy. If we can utilize this model trained in this study, we can definitely have a better chance predicting and controlling the confirm rate of the COVID-19. This study gives a more simpler choices of model to achieve such goal using transfer learning.